

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ЗАКЛАД
«ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА»

Кваліфікаційна наукова праця на правах рукопису

МАТІЄВСЬКИЙ Володимир Валерійович

УДК 378.091.33:004(043.5)

ДИСЕРТАЦІЯ

**ФОРМУВАННЯ МЕТОДОЛОГІЧНОЇ КОМПЕТЕНТНОСТІ
МАЙБУТНІХ ФАХІВЦІВ З КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ
ЗАСОБАМИ ТЕХНОЛОГІЙ
ПРОБЛЕМНО ЗОРІЄНТОВАНОГО НАВЧАННЯ**

011 Освітні, педагогічні науки

01 Освіта / Педагогіка

Подається на здобуття наукового ступеня доктора філософії

Дисертація містить результати власних досліджень. Використання ідей, результатів і текстів інших авторів мають покликання на відповідне джерело



В.В.Матієвський

Науковий керівник – **Бутенко Людмила Леонідівна**, заслужений працівник освіти України, доктор педагогічних наук, доцент, Державний заклад «Луганський національний університет імені Тараса Шевченка»

Лубни – 2026

АНОТАЦІЯ

Матієвський В.В. Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання. – Кваліфікаційна наукова праця на правах рукопису.

Дисертація на здобуття наукового ступеня доктора філософії в галузі знань 01 Освіта / Педагогіка за спеціальністю 011 Освітні, педагогічні науки. Державний заклад «Луганський національний університет імені Тараса Шевченка», м. Лубни, 2026.

Дисертаційна робота є теоретико-експериментальним дослідженням проблеми формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

Актуальність теми дослідження зумовлена необхідністю вирішення *суперечностей* між: об'єктивною потребою сучасного інформаційного суспільства та ІТ-індустрії у висококваліфікованих фахівцях з комп'ютерної інженерії, які володіють високим рівнем методологічної компетентності, здатні до системного аналізу, стратегічного мислення та розв'язання складних нестандартних завдань, і реальним станом підготовки випускників, орієнтованих переважно на вузькотехнічні, виконавські алгоритми діяльності без глибокого розуміння методологічних основ інженерії; визнанням науковою спільнотою високого дидактичного потенціалу технологій проблемно зорієнтованого навчання у розвитку дослідницьких та аналітичних навичок здобувачів вищої освіти і недостатньою теоретичною розробленістю цілісних педагогічних моделей та методичних систем, які б розкривали специфіку застосування цих технологій саме для формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії; необхідністю впровадження інноваційних технологій проблемно зорієнтованого навчання в освітній процес закладів вищої освіти для формування методологічної компетентності майбутніх фахівців ІТ-галузі та відсутністю відповідного навчально-методичного забезпечення (кейсів, проблемних завдань, критеріїв оцінювання), а також недостатньою готовністю

науково-педагогічних працівників до ефективної організації такого навчання в межах технічних дисциплін.

У дисертаційній роботі здійснено теоретичний аналіз проблеми формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання. Визначено сутність та особливості таких понять: методологія, компетентність, компетентнісний підхід, методологічна компетентність фахівців різного профілю, методологічна культура, дослідницька компетентність, методологічна компетентність майбутнього фахівця з комп'ютерної інженерії, проблемно зорієнтоване навчання, технології, форми та методи проблемно зорієнтованого навчання у вищій школі.

Методологічна компетентність майбутнього фахівця з комп'ютерної інженерії визначена як компонент професійної компетентності, динамічне інтегративне особистісне утворення, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проєктування для розв'язання складних технічних задач. Така компетентність виступає регулятивом професійної діяльності, що дозволяє фахівцям не лише адаптуватися до швидкої зміни інструментарію, а й самостійно конструювати нові алгоритми розв'язання складних науково-технічних задач на основі системного аналізу.

З урахуванням поліфункційності методологічної компетентності фахівців (когнітивна, нормативно-регулятивна, проєктно-конструктивна, рефлексивна, прогностична, преадаптивна та комунікативна функції), характеристик компетентності як динамічної інтегративної постійно збагачуваної особистісної якості фахівця визначено такі компоненти у структурі методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, як: *мотиваційний* (усвідомлення значущості фундаментальних методологічних знань та умінь для кар'єрного успіху, прагнення до інтелектуальної самореалізації в умовах стрімкого розвитку ІТ-сфери, невизначеності та нестабільності ринку праці), *когнітивний* (система

фундаментальних знань про методологію наукового пізнання, теорію систем, методологію комп'ютерно-інженерної діяльності, принципи та логіку побудови обчислювальних процесів), *праксеологічно-технологічний* (володіння загально-науковими методами пізнання, проблематизації комп'ютерно-інженерної дійсності, методами моделювання, мовами програмування, засобами автоматизації проєктування та діагностики, сформованість навичок системного мислення, толерантність до невизначеності), *рефлексивний* (здатність до самоаналізу, критичного переосмислення методів роботи та готовність до швидкої трансформації професійного досвіду в умовах невизначеності), *комунікативний* (здатність працювати в команді, здатність обирати адекватні методи, стилі та канали вербальної і письмової комунікації залежно від рівня технічної підготовки аудиторії).

Провідними методологічними підходами у формуванні методологічної компетентності майбутніх фахівців з комп'ютерної інженерії визначено системний, діяльнісний, особистісно зорієнтований, компетентнісний, праксеологічний, принципами – системності; єдності свідомості і діяльності; студентоцентризму; доповнюваності (комплементарності); випереджувальності.

За результатами аналізу вітчизняної та зарубіжної літератури представлено характеристику проблемно зорієнтованого навчання, відповідних технологій та методів. У вітчизняній психолого-педагогічній літературі усталеними є позиції щодо проблемного навчання як виду навчання, спрямованого на активізацію пізнавальних процесів, формування навичок самостійної дослідницької діяльності учнів / студентів. З 60-х рр. XX ст. у зарубіжному науковому дискурсі набуває поширення поняття «проблемно зорієнтоване навчання» (*Problem-Based Learning*) як навчальний метод, спрямований на розвиток у студентів здатності самостійно вирішувати проблеми, самостійно навчатися протягом усього життя та працювати в команді. Теоретичними засадами проблемно зорієнтованого навчання є ідеї експериментально зорієнтованого навчання Дж. Дьюї, теорії конструктивізму Ж. Піаже, соціального конструктивізму Л. Виготського, теорія практичного навчання Д. Колба, концепція рефлектуючого практика Д. Шона, концептуальні позиції реалізації проблемно зорієнтованого

навчання таких науковців, як Г. Барроуз (H. Barrows), Т. Баррет (T. Barrett), Р. Джонс (R. Jones), Х. Маурер (H. Maurer), Дж. Моуст (J. Moust), Р. Тамблін (R. Tamblyn), А. Теланг (A. Telang), С. Хмело-Сільвер (C. Hmelo-Silver).

Доведено, що у сучасній вищій освіті проблемно зорієнтоване навчання розглядається як конструктивне, спільне, контекстне та самостійне навчання. Схарактеризовано моделі/стратегії проблемно зорієнтованого навчання та відповідні форми та методи (навчання на основі лекцій із завданнями на вирішення проблем; проєктне навчання; навчання на основі кейсів та аналізу проблемних ситуацій та ін.).

З урахуванням теоретичних засад дослідження визначено критерії сформованості в майбутніх фахівців з комп'ютерної інженерії методологічної компетентності: мотиваційний, когнітивний, діяльнісний, рефлексивний, комунікативний) з відповідними показниками та рівнями – низьким, середнім, високим. Констатувальний етап експериментального дослідження на основі комплексу діагностичних методів зумовив висновок про недостатній рівень сформованості методологічної компетентності в здобувачів вищої освіти означеної спеціальності внаслідок відсутності системної інтеграції методологічних знань у зміст фахових дисциплін, фрагментарного розуміння майбутніми фахівцями взаємозв'язку між методологічними, теоретичними засадами науково-пізнавальної та професійної діяльності та відповідним інструментально-технологічним забезпеченням.

З урахуванням теоретико-методичних засад дослідження, результатів констатувального етапу експерименту визначено та обґрунтовано педагогічні умови формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії: актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну

діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності

Реалізація визначених педагогічних умов здійснювалася в процесі професійної підготовки здобувачів вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології».

Наукова новизна дослідження полягає в тому, що *вперше* визначено сутність та структуру методологічної компетентності майбутнього фахівця з комп'ютерної інженерії як компонента професійної компетентності, динамічного інтегративного особистісного утворення, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проектування для розв'язання складних технічних задач, теоретично обґрунтовано та експериментально перевірено педагогічні умови її формування засобами технологій проблемно зорієнтованого навчання (актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності);

удосконалено форми і методи професійної підготовки майбутніх фахівців з комп'ютерної інженерії з урахуванням потенціалу проблемно

зорієнтованого навчання, навчально-методичне забезпечення практико зорієнтованих форм підготовки майбутніх фахівців ІТ-сфери,

подальшого розвитку набули наукові уявлення про сутність методологічної компетентності фахівців різного профілю, методологічні та теоретичні засади використання технологій проблемно зорієнтованого навчання у вищій школі.

Практичне значення результатів дослідження визначається розробленням і впровадженням у практику роботи ЗВО, що здійснюють підготовку майбутніх фахівців з комп'ютерної інженерії, комплексів проблемних завдань, кейсів різного типу (аналітичні, техніко-діагностичні, проєктно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), завдань щодо проблематизації комп'ютерної інженерної дійсності (архітектурно-структурні, ресурсно-технологічні, інтеграційної та еволюційної сумісності, інженерної безпеки та стійкості та ін.), методичного забезпечення дослідницьких проєктів, мотиваційного тренінгу «Керуй алгоритмом: мистецтво методологічного мислення в ІТ», інтерактивного воркшопу «Ринок праці очима першокурсника: де я зараз і де хочу бути», симуляційного ІІІ-практикуму «Інженерне мислення: точка зламу», міні-хакатонів «Зворотне проєктування». Запропоновано комплекс діагностичних методик для визначення рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Ключові слова: методологія, методологічна компетентність, заклади вищої освіти, майбутні фахівці з комп'ютерної інженерії, методологічна компетентність майбутнього фахівця з комп'ютерної інженерії, проблемно зорієнтоване навчання, технології проблемно зорієнтованого навчання, командна робота.

СПИСОК ПУБЛІКАЦІЙ ЗДОБУВАЧА ЗА ТЕМОЮ ДИСЕРТАЦІЇ

Статті в наукових фахових виданнях України, зокрема, які включені до міжнародних наукометричних баз:

1. Прилипко В. М., Сохань І. В., **Матієвський В. В.** Інноваційні методики у вищій освіті як відповідь на виклики війни в Україні. *Академічні візії*. 2024. Вип. 32. DOI: <https://doi.org/10.5281/zenodo.12531404>

2. **Матієвський В. В.** Проблемно зорієнтоване навчання у вищій школі: сучасні підходи, моделі та методи. *Освіта та педагогічна наука*. 2025. № 2(189). С. 68–78. DOI: [https://doi.org/10.12958/2227-2747-2025-2\(189\)-68-78](https://doi.org/10.12958/2227-2747-2025-2(189)-68-78)

3. **Матієвський В. В.** Методологічна компетентність майбутніх фахівців з комп'ютерної інженерії: сутність, функції, структура. *Вісник науки та освіти*. 2026. № 4 (46). С. 4026–4040. DOI: [https://doi.org/10.52058/2786-6165-2026-4\(46\)-4026-4040](https://doi.org/10.52058/2786-6165-2026-4(46)-4026-4040)

4. **Матієвський В. В.** Мотиваційний компонент методологічної компетентності майбутніх фахівців з комп'ютерної інженерії: сутність та особливості формування у процесі професійної підготовки. *Педагогічна академія: наукові записки*. 2026. № 29. DOI: <https://doi.org/10.5281/zenodo.20577833>

Статті в інших виданнях, матеріали конференцій:

5. **Матієвський В. В.** Практичний досвід викладання веб розробки для фахівців інформаційно-документознавчої галузі. *Актуальні питання документознавства: історія та сьогодення* : матеріали Всеукр. наук.-практ. конф. (м. Полтава, 28 квіт. 2023 р.). Полтава : Вид-во ДЗ «ЛНУ імені Тараса Шевченка», 2023. С. 175–180. URL: <https://dspace.luguniv.edu.ua/xmlui/handle/123456789/9800>

6. Лахно В., Волошин С., Мамченко С., **Матієвський В.**, Лахно М. Реалізація на PYTHON мережі Байєса для аналізу кіберзлочинів, пов'язаних із DDOS-атаками. *Електр. фахове наук. видання «Кібербезпека: освіта, наука, техніка»*. 2024. № 4 (24). С. 161–71. DOI: <https://doi.org/10.28925/2663-4023.2024.24.161171>

7. Касаткін Д., Волошин С., Гусєв Б., **Матієвський В.** Алгоритми розробки бази знань для вдосконалення систем підтримки прийняття рішень при вирішенні

проблем кібербезпеки. *Інформаційні технології в економіці та природокористуванні*. 2024. № 1. С. 69–78. DOI: <https://doi.org/10.31548/itees.2024.01.069>

8. Лахно В., Мамченко С., Матієвський В. Аспекти виявлення кіберзагроз в мережевому трафіку університету. *Інформаційні технології в економіці та природокористуванні*. 2025. № 1 С. 73–80. DOI: <https://doi.org/10.31548/itees.2025.01.073>

9. Матієвський В. В. Аналітичні уміння та навички як складник методологічної компетентності майбутніх фахівців з комп'ютерної інженерії. *Інноваційні тенденції сьогодення в сфері природничих, гуманітарних та точних наук*: збірник наук. праць з матеріалами VIII Міжнар. наук. конф., м. Тернопіль, 20 лютого, 2026 р. / Міжнар. центр наук. досліджень. Вінниця : ТОВ «УКРЛОГОС Груп», 2026. С. 247–251. URL: <https://archives.mcnd.org.ua/index.php/conference-proceeding/issue/view/20.02.2026/82>

10. Матієвський В. В. Використання кейс-методу як інструменту проблемно зорієнтованого навчання у підготовці майбутніх фахівців з комп'ютерної інженерії. *Теорія і практика сучасної науки та освіти*: матеріали XVIII Міжнар. наук.-практ. конф. м. Львів, 9-10 березня 2026 року. Львів: Львівський науковий форум, 2026. С. 13–16. URL: <http://lviv-forum.inf.ua/save/2026/9-10.03/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA.pdf>

11. Матієвський В. В. Методологічна компетентність майбутніх фахівців з комп'ютерної інженерії в умовах цифровізації освіти. *Плекаємо особистість*: наук.-метод. альманах. Вип. 11. Лубни: Держ. закл. «Луган. нац. ун-т імені Тараса Шевченка», 2026. С.110–114. URL: <https://drive.google.com/drive/u/0/folders/1pz8ZU5yIrWFBq98a9NslJJbpPZIaaeej>

12. Матієвський В. В. Форми та методи проблемно зорієнтованого навчання у професійній підготовці майбутніх фахівців з комп'ютерної інженерії. *Recent Trends in Science: Proceedings of the 5th International Scientific and Practical Internet Conference*, May 14-15. Dnipro: FOP Marenichenko V.V., 2026. P. 171–173. URL: <http://www.wayscience.com/wp-content/uploads/2026/05/Conference-Proceedings-May-14-15-2026.pdf>

13. Матієвський В. В. Преадаптаційна функція методологічної підготовки майбутніх фахівців з комп'ютерної інженерії. *Наука і освіта в глобальному та національному вимірах: виклики, загрози, перспективи розвитку* : зб. наук. праць за матер. V міжнародної наук.-практ. конф. (Лубни, 28-29 травня 2026 р.) / за ред. Я. І. Юрків, Н. О. Островської. Лубни : ДЗ «Луганський національний університет імені Тараса Шевченка», 2026. С. 111–114. URL: <https://drive.google.com/drive/folders/1mY6Sa6ALP4At2TF6HHZwFoG-W1rKNyvX>

Навчальний посібник

14. Коваленко О.Є., Волошин С.М., Гусєв Б.С., Нікітенко Є.В., Матієвський В. В. Комп'ютерна схемотехніка. Частина 2 : навчальний посібник. Київ : Компринт, 2023. 331 с.

ABSTRACT

Matiievskiy V.V. Formation of Methodological Competence of Future Computer Engineering Specialists by Means of Problem-Based Learning Technologies. – Thesis submitted as a manuscript for the degree of Doctor of Philosophy.

Dissertation for the degree of Doctor of Philosophy in the field of knowledge 01 Education / Pedagogy, specialty 011 Education, Pedagogical Sciences. State Institution «Taras Shevchenko Luhansk National University», Lubny, 2026.

The dissertation is a theoretical and experimental study of the problem of forming the methodological competence of future computer engineering specialists by means of problem-based learning technologies.

The relevance of the research topic is determined by the need to resolve the *contradictions* between: the objective need of the modern information society and the IT industry for highly qualified computer engineering specialists who possess a high level of methodological competence, are capable of systems analysis, strategic thinking, and solving complex non-standard tasks, and the actual state of training of graduates, who are mainly focused on narrow-technical, executive algorithms of activity without a deep understanding of the methodological foundations of engineering; the scientific

community's recognition of the high didactic potential of problem-based learning technologies in the development of research and analytical skills of higher education learners, and the insufficient theoretical development of holistic pedagogical models and methodological systems that would reveal the specifics of applying these technologies precisely for the formation of the methodological competence of future computer engineering specialists; the need to introduce innovative problem-based learning technologies into the educational process of higher education institutions to form the methodological competence of future IT specialists, and the lack of appropriate educational and methodological support (cases, problem tasks, assessment criteria), as well as the insufficient readiness of academic staff to effectively organize such learning within technical disciplines.

The dissertation provides a theoretical analysis of the problem of forming the methodological competence of future computer engineering specialists by means of problem-based learning technologies. The essence and features of the following concepts are defined: methodology, competence, competence-based approach, methodological competence of specialists in various profiles, methodological culture, research competence, methodological competence of a future computer engineering specialist, problem-based learning, technologies, forms, and methods of problem-based learning in higher education.

The methodological competence of a future computer engineering specialist is defined as a component of professional competence, a dynamic integrative construct of the individual's personality that determines the engineer's ability to consciously choose, combine, and effectively apply fundamental methods of scientific cognition, mathematical modeling, and systems design to solve complex technical tasks. Such competence acts as a regulator of professional activity, allowing specialists not only to adapt to rapid changes in tools but also to independently construct new algorithms for solving complex scientific and technical tasks based on systems analysis.

Taking into account the multifunctionality of specialists' methodological competence (cognitive, normative-regulatory, design-constructive, reflective,

prognostic, pre-adaptive, and communicative functions), and the characteristics of competence as a dynamic, integrative, constantly enriched personal quality of a specialist, the following components in the structure of the methodological competence of future computer engineering specialists are determined: *motivational* (awareness of the importance of fundamental methodological knowledge and skills for career success, striving for intellectual self-realization in the conditions of rapid development of the IT sphere, uncertainty, and instability of the labor market), *cognitive* (a system of fundamental knowledge about the methodology of scientific cognition, systems theory, methodology of computer engineering activity, principles and logic of building computational processes), *praxeological-technological* (mastery of general scientific methods of cognition, problematization of computer engineering reality, modeling methods, programming languages, computer-aided design and diagnostic tools, formed systems thinking skills, tolerance for uncertainty), *reflective* (the ability for self-analysis, critical rethinking of working methods, and readiness for rapid transformation of professional experience in conditions of uncertainty), *communicative* (the ability to work in a team, the ability to choose adequate methods, styles, and channels of verbal and written communication depending on the level of technical training of the audience).

The leading methodological approaches in forming the methodological competence of future computer engineering specialists are defined as systemic, activity-based, personality-oriented, competence-based, and praxeological; the principles are systemicity; unity of consciousness and activity; student-centeredness; complementarity; and proactivity.

Based on the results of the analysis of domestic and foreign literature, the characteristics of problem-based learning, relevant technologies, and methods are presented. In domestic psychological and pedagogical literature, there are established positions regarding problem-based learning as a type of learning aimed at activating cognitive processes and forming independent research skills of pupils/students. Since the 1960s, the concept of "*Problem-Based Learning*" has become widespread in foreign scientific discourse as an educational method aimed at developing students' ability to

independently solve problems, learn continuously throughout life, and work in a team. The theoretical foundations of problem-based learning are J. Dewey's ideas of experientially-oriented learning, J. Piaget's constructivism theories, L. Vygotsky's social constructivism, D. Kolb's experiential learning theory, D. Schön's concept of the reflective practitioner, and the conceptual positions on implementing problem-based learning by scientists such as H. Barrows, T. Barrett, R. Jones, H. Maurer, J. Moust, R. Tamblyn, A. Telang, and C. Hmelo-Silver.

It is proven that in modern higher education, problem-based learning is considered as constructive, collaborative, contextual, and self-directed learning. Models/strategies of problem-based learning and corresponding forms and methods (problem-based lecture learning; project-based learning; case-based learning and problem situation analysis, etc.) are characterized.

Taking into account the theoretical foundations of the research, the criteria for the formation of methodological competence in future computer engineering specialists are determined: motivational, cognitive, activity-based, reflective, communicative, with corresponding indicators and levels – low, medium, and high. The ascertaining stage of the experimental research, based on a set of diagnostic methods, led to the conclusion about an insufficient level of formation of methodological competence in higher education learners of this specialty due to the lack of systemic integration of methodological knowledge into the content of professional disciplines, and the fragmentary understanding by future specialists of the relationship between the methodological and theoretical foundations of scientific-cognitive and professional activities and the corresponding instrumental and technological support.

Taking into account the theoretical and methodological foundations of the research and the results of the ascertaining stage of the experiment, the pedagogical conditions for forming the methodological competence of future computer engineering specialists are defined and substantiated: activation of motivation and awareness of the value of methodological knowledge and skills in the context of professional development and career growth of future specialists; integration of fundamental methodological knowledge with profile computer engineering disciplines based on

interdisciplinary problem tasks; inclusion of future computer engineering specialists in quasi-professional activities based on various types of cases, problematisation tasks in computer engineering, and mini-hackathons aimed at developing methodological skills and teamwork competencies; development of the reflective stance of future specialists at all stages of forming methodological competence.

The implementation of the defined pedagogical conditions was carried out in the process of professional training of higher education learners majoring in specialty 123 "Computer Engineering", knowledge domain 12 "Information Technologies".

The scientific novelty of the research lies in the fact that *for the first time*, the essence and structure of the methodological competence of a future computer engineering specialist are defined as a component of professional competence, a dynamic integrative construct of the individual's personality that determines the engineer's ability to consciously choose, combine, and effectively apply fundamental methods of scientific cognition, mathematical modeling, and systems design to solve complex technical tasks; the pedagogical conditions for its formation by means of problem-based learning technologies are theoretically substantiated and experimentally verified (activation of motivation and awareness of the value of methodological knowledge and skills in the context of professional development and career growth of future specialists; integration of fundamental methodological knowledge with profile computer engineering disciplines based on interdisciplinary problem tasks; inclusion of future computer engineering specialists in quasi-professional activities based on various types of cases, problematisation tasks in computer engineering, and mini-hackathons aimed at developing methodological skills and teamwork competencies; development of the reflective stance of future specialists at all stages of forming methodological competence;

the forms and methods of professional training of future computer engineering specialists *have been improved* taking into account the potential of problem-based learning, as well as the educational and methodological support of practice-oriented forms of training for future IT specialists;

scientific concepts about the essence of the methodological competence of specialists in various profiles, and the methodological and theoretical foundations of using problem-based learning technologies in higher education *have been further developed*.

The practical significance of the research results is determined by the development and introduction into the practice of higher education institutions that train future computer engineering specialists, complexes of problem tasks, cases of various types (analytical, technical-diagnostic, design-constructive, situational-strategic, cases of retrospective analysis), tasks regarding the problematization of computer engineering reality (architectural-structural, resource-technological, integration and evolutionary compatibility, engineering security and sustainability, etc.), methodological support for research projects, motivational training "Control the algorithm: the art of methodological thinking in IT", interactive workshop "The labor market through the eyes of a freshman: where I am now and where I want to be", simulation AI-practicum "Engineering thinking: the breaking point", and mini-hackathons "Reverse engineering". A complex of diagnostic methods is proposed to determine the level of formation of the methodological competence of future computer engineering specialists.

Keywords: methodology, methodological competence, higher education institutions, future computer engineering specialists, methodological competence of a future computer engineering specialist, problem-based learning, problem-based learning technologies, teamwork.

LIST OF THE APPLICANT'S PUBLISHED PAPERS

Scientific papers in professional scientific editions of Ukraine:

1. Prylypko, V. M., Sokhan, I. V., **Matiievskyi, V. V.** (2024). Innovatsiini metodyky u vyshchii osviti yak vidpovid na vyklyky viiny v Ukraini [Innovative Methodologies in Higher Education as a Response to the Challenges of War in Ukraine]. *Akademichni vizii – Academic Visions*, (32). DOI: <https://doi.org/10.5281/zenodo.12531404> [in Ukrainian].

2. **Matiiievskiy, V. V.** (2025). Problemno zoriiientovane navchannia u vyshchii shkoli: suchasni pidkhody, modeli ta metody [Problem-based Learning in Higher Education: Modern Approaches, Models, and Methods]. *Osvita ta pedahohichna nauka – Education and Pedagogical Sciences*, 2(189), 68–78. DOI: [https://doi.org/10.12958/2227-2747-2025-2\(189\)-68-78](https://doi.org/10.12958/2227-2747-2025-2(189)-68-78) [in Ukrainian].

3. **Matiiievskiy, V. V.** (2026). Metodolohichna kompetentnist maibutnikh fakhivtsiv z kompiuternoi inzhenerii: sutnist, funktsii, struktura [Methodological competence of future computer engineering specialists: essence, functions, structure]. *Visnyk nauky ta osvity – Bulletin of Science and Education*, 4(46), 4026–4040. DOI: [https://doi.org/10.52058/2786-6165-2026-4\(46\)-4026-4040](https://doi.org/10.52058/2786-6165-2026-4(46)-4026-4040) [in Ukrainian].

4. **Matiiievskiy, V. V.** (2026). Motyvatsiinyi komponent metodolohichnoi kompetentnosti maibutnikh fakhivtsiv z kompiuternoi inzhenerii: sutnist ta osoblyvosti formuvannia u protsesi profesiinoi pidhotovky [Motivational component of methodological competence of future computer engineering specialists: essence and features of formation in the process of professional training]. *Pedahohichna akademiia: naukovy zapysky – Pedagogical Academy: Scientific Notes*, (29). DOI: <https://doi.org/10.5281/zenodo.20577833> [in Ukrainian].

Scientific papers certifying the approbation of the dissertation materials:

5. **Matiiievskiy, V. V.** (2023). Praktychnyi dosvid vykladannia veb-rozrobky dlia fakhivtsiv informatsiino-dokumentoznavchoi haluzi [Practical experience of teaching web development for specialists in the information and document science field]. *Aktualni pytannia dokumentoznavstva: istoriia ta siohodennia – Topical Issues of Document Science: History and Present: Proceedings from Vseukr. nauk.-prakt. konferentsiia* (pp. 175–180). Poltava: Vyd-vo DZ «LNU imeni Tarasa Shevchenka» [in Ukrainian]. URL: <https://dspace.luguniv.edu.ua/xmlui/handle/123456789/9800>

6. Lakhno, V., Voloshyn, S., Mamchenko, S., **Matiiievskiy, V.**, Lakhno, M. (2024). Realizatsiia na PYTHON merezhi Baiesa dlia analizu kiberzlochyniv, poviazanykh iz DDOS-atakamy [Implementation of a Bayesian network in Python for the analysis of cybercrimes related to DDoS attacks]. *Kiberbezpeka: osvita,*

nauka, tekhnika – Cybersecurity: Education, Science, Technique, 4(24), 161–171. DOI: <https://doi.org/10.28925/2663-4023.2024.24.161171> [in Ukrainian].

7. Kasatkin, D., Voloshyn, S., Husiev, B., **Matiievskiy, V.** (2024). Alhorytmy rozrobky bazy znan dla vdoskonalennia system pidtrymky pryiniattia rishen pry vyrishenni problem kiberbezpeky [Knowledge base development algorithms for improving decision support systems in solving cybersecurity problems]. *Informatsiini tekhnolohii v ekonomitsi ta pryrodo-korystuvanni – Information Technologies in Economy and Environmental Management*, (1), 69–78. DOI: <https://doi.org/10.31548/itees.2024.01.069> [in Ukrainian].

8. Lakhno, V., Mamchenko, S., **Matiievskiy, V.** (2025). Aspekty vyiavlennia kiberzahroz v merezhevomu trafiku universytetu [Aspects of cyber threat detection in university network traffic]. *Informatsiini tekhnolohii v ekonomitsi ta pryrodokorystuvanni – Information Technologies in Economy and Environmental Management*, (1), 73–80. DOI: <https://doi.org/10.31548/itees.2025.01.073> [in Ukrainian].

9. **Matiievskiy, V. V.** (2026). Analitychni uminnia ta navychky yak skladnyk metodolohichnoi kompetentnosti maibutnikh fakhivtsiv z kompiuternoi inzhenerii [Analytical skills and abilities as a component of methodological competence of future computer engineering specialists]. *Innovatsiini tendentsii siohodennia v sferi pryrodnychkh, humanitarnykh ta tochnykh nauk – Innovative Trends of Today in the Field of Natural, Humanitarian and Exact Sciences: Proceedings from VIII Mizhnarodna naukova konferentsiia* (pp. 247–251). Vinnytsia: TOV «UKRLOHOS Hrup» [in Ukrainian]. URL: <https://archives.mcnd.org.ua/index.php/conference-proceeding/issue/view/20.02.2026/82>

10. **Matiievskiy, V. V.** (2026). Vykorystannia keis-metodu yak instrumentu problemno zoriietovanoho navchannia u pidhotovtsi maibutnikh fakhivtsiv z kompiuternoi inzhenerii [Use of the case method as a tool of problem-based learning in the training of future computer engineering specialists]. *Teoriia i praktyka suchasnoi nauky ta osvity – Theory and Practice of Modern Science and Education: Proceedings from XVIII Mizhnarodna naukovo-praktychna konferentsiia* (pp. 13–16). Lviv: Lvivskiyi naukovyi forum [in Ukrainian]. URL:

<http://lviv-forum.inf.ua/save/2026/9->

[10.03/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA.pdf](http://lviv-forum.inf.ua/save/2026/9-10.03/%D0%97%D0%B1%D1%96%D1%80%D0%BD%D0%B8%D0%BA.pdf)

11. **Matiievskiy, V. V.** (2026). Metodolohichna kompetentnist maibutnikh fakhivtsiv z kompiuternoї inzhenerii v umovakh tsyfrovizatsii osvity [Methodological competence of future computer engineering specialists in the conditions of digitalization of education]. *Plekaemo osobystist: nauk.-metod. almanakh – Nurturing Personality: Scientific and Methodological Almanac*, (11), 110–114. Lubny: Derzh. zakl. «Luhan. nats. un-t imeni Tarasa Shevchenka» [in Ukrainian]. URL: <https://drive.google.com/drive/u/0/folders/1pz8ZU5yIrWFBq98a9NslJJbpPZIaaej>

12. **Matiievskiy, V. V.** (2026). Formy ta metody problemno zoriietovanoho navchannia u profesiinii pidhotovtsi maibutnikh fakhivtsiv z kompiuternoї inzhenerii [Forms and methods of problem-based learning in the professional training of future computer engineering specialists]. *Recent Trends in Science: Proceedings from 5th International Scientific and Practical Internet Conference* (pp. 171–173). Dnipro: FOP Marenichenko V.V. [in Ukrainian]. URL: <http://www.wayscience.com/wp-content/uploads/2026/05/Conference-Proceedings-May-14-15-2026.pdf>

13. **Matiievskiy, V. V.** (2026). Preadaptatsiina funktsiia metodolohichnoi pidhotovky maibutnikh fakhivtsiv z kompiuternoї inzhenerii [Preadaptation function of methodological training of future computer engineering specialists]. *Nauka i osvita v hlobalnomu ta natsionalnomu vymirakh: vyklyky, zahrozy, perspektyvy rozvytku – Science and Education in Global and National Dimensions: Challenges, Threats, Prospects of Development: Proceedings from V Mizhnarodna naukovo-praktychna konferentsiia* (pp. 111–114). Lubny: DZ «Luhanskyi natsionalnyi universytet imeni Tarasa Shevchenka» [in Ukrainian]. URL: <https://drive.google.com/drive/folders/1mY6Sa6ALP4At2TF6HHZwFoG-W1rKNyvX>

Study guide

Kovalenko, O. Ye., Voloshyn, S. M., Husiev, B. S., Nikitenko, Ye. V., **Matiievskiy, V. V.** (2023). *Kompiuterna skhemotekhnika. Chastyna 2: navchalnyi posibnyk* [Computer circuitry. Part 2: study guide]. Kyiv: Komprynt [in Ukrainian].

ЗМІСТ

АНОТАЦІЯ	2
ВСТУП.....	20
РОЗДІЛ 1. Теоретико-методичні засади формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії	32
1.1. Поняття «методологічна компетентність» у сучасному соціогуманітарному дискурсі	32
1.2. Сутність та структура методологічної компетентності майбутніх фахівців з комп'ютерної інженерії	56
1.3. Технології проблемно зорієнтованого навчання у вищій школі: сутність, моделі, форми та методи	86
1.4. Діагностика рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії	110
ВИСНОВКИ ДО РОЗДІЛУ 1	135
РОЗДІЛ 2. Експериментальне дослідження педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання	139
2.1. Обґрунтування педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно-орієнтованого навчання	139
2.2. Упровадження педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії	176
2.3. Аналіз результатів дослідно-експериментальної роботи	204
ВИСНОВКИ ДО РОЗДІЛУ 2	226
ВИСНОВКИ	228
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	233
ДОДАТКИ	256

ВСТУП

Актуальність дослідження проблеми формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії зумовлена стрімким науково-технічним прогресом та неперервною трансформацією галузі інформаційних технологій. У сучасних умовах цифровізації суспільства традиційний підхід до підготовки інженерів, орієнтований лише на засвоєння конкретних мов програмування чи інструментальних засобів, швидко втрачає свою ефективність, оскільки технологічні рішення у сфері ІТ оновлюються швидше, ніж завершується цикл навчання у закладі вищої освіти. За таких обставин виникає гостра потреба у фахівцях з комп'ютерної інженерії, які володіють не просто набором готових алгоритмів, а фундаментальними методологічними знаннями, що дозволяють системно мислити, самостійно опановувати інновації, критично аналізувати складні процеси та ефективно вирішувати нестандартні інженерні завдання. За даними Асоціації ІТ України кількість фахівців ІТ-сфери за останні 5 років зросла на 20,05% і стала складати приблизно 305 тис. осіб [51, с. 27]. У сучасних умовах професійна підготовка ІТ-фахівців набула стратегічного значення для обороноздатності та економіки України, оскільки ІТ-сектор масштабно залучений до створення новітніх технологічних рішень.

Комп'ютерна інженерія – це «предметна область, що об'єднує науку та технологію проектування, побудови, впровадження та обслуговування програмних та апаратних компонентів сучасних обчислювальних систем та комп'ютерного обладнання» [106, с. 5]. Професійна підготовка фахівців з комп'ютерної інженерії здійснюється відповідно до Стандарту вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти [143] (з 2024 р. відповідно до Міжнародної стандартної класифікації освіти ISCED F – галузь знань «Інформаційні технології», спеціальність – F 7 Комп'ютерна інженерія). Підготовку зазначених фахівців орієнтована на вимоги *Computing Curricula 2020*

– спільної міжнародної ініціативи провідних професійних організацій ACM та IEEE-CS, яка визначає сучасні рамкові стандарти та рекомендації для вищої освіти у галузі комп'ютерних наук та інформаційних технологій [180] та *Subject Benchmark Statement. Computing, 2022* – стандарту Агенції із забезпечення якості вищої освіти (QAA – Quality Assurance Agency for Higher Education), який визначає очікувані результати навчання, академічні стандарти та ключові компетенції для випускників бакалаврських та магістерських програм у галузі комп'ютерних наук та інформаційних технологій [213].

Формування методологічної компетентності виступає надійним базисом для розвитку здатності майбутніх фахівців з комп'ютерної інженерії здійснювати науково-дослідну, проєктувальну та аналітичну діяльність на вищому рівні абстракції, інтегрує знання про принципи організації, методи та засоби діяльності у сфері комп'ютерингу, дозволяє фахівцю успішно адаптуватися до умов невизначеності та динамічних змін ринку праці.

Важливим напрямом модернізації процесу професійної підготовки майбутніх фахівців ІТ-сфери є упровадження технологій проблемно зорієнтованого навчання. Завдяки зануренню в проблемні ситуації в студентів розвивається критичне та системне мислення, що дозволяє їм не просто писати код чи проєктувати апаратне забезпечення за шаблоном, а глибоко розуміти суть проблеми, аналізувати причинно-наслідкові зв'язки та пропонувати оптимальні інженерні рішення. Процес пошуку виходу з проблемної ситуації стимулює дослідницьку активність майбутніх фахівців, змушує самостійно шукати інформацію, освоювати нові інструменти та технології, що закладає надійний фундамент для концепції навчання впродовж усього життя. Крім того, такий підхід максимально наближає академічне середовище до умов реального ІТ-ринку. Студенти вчаться працювати в умовах невизначеності, коли початкові вимоги до проєкту можуть бути розмитими або суперечливими, що є типовим для інженерної практики. Спільне вирішення комплексних завдань у команді допомагає майбутнім комп'ютерним інженерам опанувати життєво важливі

гнучкі навички, серед яких ефективна комунікація, розподіл обов'язків, аргументація власної технічної позиції та управління часом.

Питаннями компетентнісного підходу до навчання, змісту і структури компетенцій та компетентностей здобувачів освіти різного рівня присвячено праці таких дослідників, як І. Бех, Н. Бібік, О. Жукова, В. Луговий, О. Овчарук, С. Паламар, О. Пометун, С. Сисоєва та ін.

Тенденції розвитку ІТ-освіти в Україні репрезентовано у дослідженні М.-О. Єршова, теоретико-методичні засади підготовки фахівців ІТ-сфери – у працях О. Баранюк, І. Гарко, С. Литвинової, Н. Морзе, Л. Ожиндович, С. Петренка, С. Проскури, М. Семенова, С. Семерікова, Я. Сікори та ін. Проблеми підготовки майбутніх фахівців ІТ-галузі у зарубіжних університетах висвітлено у наукових розвідках О. Кін та М. Якімової, Я. Сікори.

Загальні питання формування професійної компетентності майбутніх бакалаврів з комп'ютерної інженерії представлено у наукових розвідках таких дослідників, як М. Бирка, Л. Макаренко, І. Синельник, О. Синельник, Л. Слободянюк, М. Сорокопуд, К. Стрюк, М. Чумак, Ю. Шпильовий, які акцентують увагу на необхідності формування в майбутніх фахівців досліджуваної галузі системного та критичного мислення, аналітичних та рефлексивних навичок.

Зарубіжними дослідниками розглядається комплекс питань у контексті методологічної підготовки фахівців ІТ-галузі. Ф. Ашенбергер (F. Aschenberger) та Т. Пффеффер (T. Pfeffer) акцентують увагу на дослідницькій грамотності та процедурному знанні, М. Менекше (M. Menekse) характеризує сутнісні ознаки метакогнітивної рефлексії, Ж. Вінг (J. Wing) – обчислювального мислення, Г. Мадхаван (G. Madhavan) – інженерного мислення як провідної якості фахівців технічної галузі, М. Боррего (M. Borrego) та Й. Бернхард (J. Bernhard) – міждисциплінарних аспектів інженерної освіти. У фундаментальній праці Л. Флоріді (L. Floridi) відзначені положення щодо формування вмінь фахівців у галузі комп'ютерної інженерії аналізувати складні системи через «рівні абстракції» (*Levels of Abstraction, LoA*).

Концептуальні засади методології в контексті наукової та професійної діяльності досліджували С. Гончаренко, М. Кубаєвський, А. Фурман та ін. Значення методологічних регулятивів як інструменту не лише теоретичної, а й практико перетворювальної діяльності розкрито у працях Л. Бутенко, О. Гречаного, В. Сабадухи обґрунтовують взаємозв'язок освоєння методології як системи та інтелектуальних, рефлексивних здібностей майбутніх фахівців.

Загальні позиції щодо методологічної підготовки здобувачів вищої освіти репрезентовано у наукових працях Л. Ваховського, А. Вихруща, Л. Ковальчук, В. Кравцова, С. Сисоєвої, Я. Топольник, А. Фурмана та ін. Сутнісні ознаки методологічної компетентності фахівців різного профілю розкрито у наукових розвідках вітчизняних дослідників (І. Козубцов, В. Кравцов, Н. Кугай, О. Лаврентьєва, О. Малихін, Л. Нічуговська, М. Опачко, О. Савицька, С. Шехавцова та ін.).

Теоретико-методичні аспекти проблемного, проблемно зорієнтованого навчання в освіті в цілому та у вищій освіті зокрема висвітлено у наукових працях таких вітчизняних дослідників, як Л. Козуля, О. Лапузіна, Г. Луценко, В. Павленко, О. Романовський, Т. Семигіна. Питання проблемно зорієнтованого навчання у контексті професійної підготовки майбутніх фахівців з програмної інженерії розглядає О. Баранюк, фахівців медичної сфери – М. Авраменко, О. Беляєва, Л. Горяча, Ю. Колесник, Л. Хламанова, Л. Хмель, Ю. Чайковський та ін. У наукових працях зарубіжних учених – Г. Барроуз (H. Barrows), Т. Баррет (T. Barrett), Р. Джонс (R. Jones), Х. Маурер (H. Maurer), Дж. Моуст (J. Moust), Р. Тамблін (R. Tamblyn), А. Теланг (A. Telang), С. Хмело-Сільвер (C. Hmelo-Silver) та ін. – розглядаються сутність, моделі та методи проблемно зорієнтованого навчання (*Problem Based Learning*).

У дисертаційних роботах висвітлено різні аспекти формування методологічної компетентності та методологічної культури фахівців різного профілю, Так, Лі Ліцюань досліджував питання формування методологічної компетентності майбутніх учителів музичного мистецтва в процесі вокальної підготовки [75], О. Лаврентьєва – теоретичні і методичні засади розвитку

методологічної культури майбутніх учителів природничих дисциплін у процесі професійної підготовки [70], Н. Кугай – формування методологічних знань і вмінь майбутніх учителів математики у процесі навчання дисциплін математичного циклу [65]. Питання формування дослідницької компетентності майбутніх викладачів вищих навчальних закладів в умовах магістратури репрезентовано у дисертації Л. Бондаренко [8], науково-дослідницької компетентності майбутніх інженерів-програмістів в умовах освітнього середовища вищого навчального закладу – М. Вінника [17].

У дисертаційних роботах представлено різні аспекти професійної підготовки майбутніх фахівців ІТ-сфери в цілому та майбутніх фахівців з комп'ютерної інженерії зокрема. Л. Зубик досліджувала формування професійних компетентностей майбутніх бакалаврів з інформаційних технологій у процесі вивчення фахових дисциплін [39], В. Концедайло – застосування ігрових симуляторів у формуванні професійних компетентностей майбутніх інженерів-програмістів [59], В. Круглик – систему підготовки майбутніх інженерів-програмістів до професійної діяльності у вищих навчальних закладах [62], Ю. Сабодош – розвиток самоосвітньої компетентності майбутніх інженерів комп'ютерних систем і автоматики [121], Я. Сікора – теоретико-методичні засади адаптивної системи професійної підготовки майбутніх фахівців з інформаційних технологій в умовах цифровізації [136], Л. Слободянюк – формування професійної компетентності майбутніх бакалаврів з комп'ютерної інженерії у закладах фахової передвищої освіти [138], К. Стрюк – формування професійної компетентності молодших спеціалістів з комп'ютерної інженерії у радіотехнічних коледжах [145].

Аналіз наукової та навчально-методичної літератури засвідчив, що питання формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії на основі проблемно зорієнтованого навчання ще не було предметом окремого дослідження.

Актуальність теми дослідження зумовлена необхідністю вирішення *суперечностей* між: об'єктивною потребою сучасного інформаційного

суспільства та IT-індустрії у висококваліфікованих фахівцях з комп'ютерної інженерії, які володіють високим рівнем методологічної компетентності, здатні до системного аналізу, стратегічного мислення та розв'язання складних нестандартних завдань, і реальним станом підготовки випускників, орієнтованих переважно на вузькотехнічні, виконавські алгоритми діяльності без глибокого розуміння методологічних основ інженерії; визнанням науковою спільнотою високого дидактичного потенціалу технологій проблемно зорієнтованого навчання у розвитку дослідницьких та аналітичних навичок здобувачів вищої освіти і недостатньою теоретичною розробленістю цілісних педагогічних моделей та методичних систем, які б розкривали специфіку застосування цих технологій саме для формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії; необхідністю впровадження інноваційних технологій проблемно зорієнтованого навчання в освітній процес закладів вищої освіти для формування методологічної компетентності майбутніх фахівців IT-галузі та відсутністю відповідного навчально-методичного забезпечення (кейсів, проблемних завдань, критеріїв оцінювання), а також недостатньою готовністю науково-педагогічних працівників до ефективної організації такого навчання в межах технічних дисциплін.

Таким чином, актуальність проблеми дослідження, наявні суперечності та недостатній рівень теоретичної й практичної розробленості обумовили вибір теми дисертаційної роботи **«Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання»**.

Зв'язок роботи з науковими програмами, планами, темами. Дослідження виконане відповідно до наукової теми кафедри педагогіки ДЗ «Луганський національний університет імені Тараса Шевченка»: «Якість сучасної освіти: теорія, моніторинг, технології» (номер держреєстрації 0117U002854). Тему роботи затверджено Вченою радою ДЗ «Луганський національний університет імені Тараса Шевченка» (протокол № 11 від 30.05.2025 р.).

Об'єкт дослідження – процес професійної підготовки майбутніх фахівців з комп'ютерної інженерії.

Предмет дослідження – педагогічні умови формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

Мета дослідження – теоретично обґрунтувати та експериментально перевірити ефективність педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

Гіпотеза дослідження ґрунтується на припущенні про те, що формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання буде ефективним за таких умов: актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності.

Відповідно до предмета й мети визначено такі **завдання** дослідження:

1. На основі аналізу філософської, психолого-педагогічної літератури схарактеризувати наукові підходи до сутності поняття «методологічна компетентність».

2. Визначити сутність і структурні компоненти методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

3. Розкрити потенціал технологій проблемно зорієнтованого навчання у процесі формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

4. Визначити критерії та показники сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

5. Теоретично обґрунтувати педагогічні умови формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

6. Експериментально перевірити ефективність педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

Теоретико-методологічну основу дослідження становлять: наукові положення системного (С. Гончаренко, В. Кушнір, І. Малафіїк), діяльнісного (І. Бех, С. Максименко, Л. Шелестова), особистісно зорієнтованого (І. Бех, О. Дубасенюк, О. Пехота), компетентнісного (І. Драч, О. Локшина, В. Луговий, О. Овчарук), праксеологічного (Т. Котарбінський) методологічних підходів; концептуальні засади методології наукової та практичної діяльності (Л. Бутенко, Л. Ваховський, С. Гончаренко, Г. Гречаний, М. Кубаєвський, А. Фурман); теоретико-методичні засади професійної підготовки майбутніх фахівців ІТ-сфери в цілому та майбутніх фахівців з комп'ютерної інженерії зокрема (М.-О. Єршов, Л. Макаренко, Н. Морзе, С. Петренко, Я. Сікора, М. Семенов, С. Семеріков, Л. Слободянюк, К. Стрюк, М. Чумак, Ю. Шпильовий; Ф. Ашенбергер (F. Aschenberger), Т. Пфеффер (T. Pfeffer), Ж. Вінг (J. Wing), Г. Мадхаван (G. Madhavan), М. Боррего (M. Borrego) та Й. Бернхард (J. Bernhard), Л. Флоріді (L. Floridi)), формування методологічної компетентності фахівців різного профілю (І. Козубцов, В. Кравцов, Н. Кугай, О. Лаврентьєва, Л. Нічуговська, М. Опачко, Лі Цюань, О. Малихін, Б. Рубінська); теорія і практики проблемно зорієнтованого навчання у вищій школі (Л. Козуля, О. Лапузіна, Г. Луценко, В. Павленко, О. Романовський, Т. Семігіна, Г. Барроуз (H. Barrows), Т. Баррет (T. Barrett),

Р. Джонс (R. Jones), Х. Маурер (H. Maurer), Дж. Моуст (J. Moust), Р. Тамблін (R. Tamblyn), А. Теланг (A. Telang), С. Хмело-Сільвер (C. Hmelo-Silver)).

Для вирішення поставлених завдань, досягнення мети використано комплекс **методів** дослідження: теоретичних – аналіз філософської, психолого-педагогічної літератури для визначення теоретико-методологічних засад дисертаційної праці; систематизація й класифікація наукових положень, синтез, узагальнення для обґрунтування понятійно-категоріального апарату дослідження, визначення особливостей формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованого навчання; узагальнення й систематизація для обґрунтування відповідних педагогічних умов; *емпіричних* – анкетування, тестування, експертна оцінка та самооцінка для виявлення рівня сформованості методологічної компетентності студентів спеціальності «Комп'ютерна інженерія»; педагогічний експеримент для перевірки ефективності педагогічних умов; *методи математичної статистики* для визначення статистичної значущості отриманих результатів.

Наукова новизна дослідження полягає в тому, що *вперше* визначено сутність та структуру методологічної компетентності майбутнього фахівця з комп'ютерної інженерії як компонента професійної компетентності, інтегративне утворення особистості, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проєктування для розв'язання складних технічних задач, теоретично обґрунтовано та експериментально перевірено педагогічні умови її формування засобами технологій проблемно зорієнтованого навчання (актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань

щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності);

удосконалено форми і методи професійної підготовки майбутніх фахівців з комп'ютерної інженерії з урахуванням потенціалу проблемно зорієнтованого навчання, навчально-методичне забезпечення практико зорієнтованих форм підготовки майбутніх фахівців ІТ-сфери,

подальшого розвитку набули наукові уявлення про сутність методологічної компетентності фахівців різного профілю, методологічні та теоретичні засади використання технологій проблемно зорієнтованого навчання у вищій школі.

Практичне значення результатів дослідження визначається розробленням і впровадженням у практику роботи ЗВО, що здійснюють підготовку майбутніх фахівців з комп'ютерної інженерії, комплексів проблемних завдань, кейсів різного типу (аналітичні, техніко-діагностичні, проєктно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), завдань щодо проблематизації комп'ютерної інженерної дійсності (архітектурно-структурні, ресурсно-технологічні, інтеграційної та еволюційної сумісності, інженерної безпеки та стійкості та ін.), методичного забезпечення дослідницьких проєктів, мотиваційного тренінгу «Керуй алгоритмом: мистецтво методологічного мислення в ІТ», інтерактивного воркшопу «Ринок праці очима першокурсника: де я зараз і де хочу бути», симуляційного ІІІ-практикуму «Інженерне мислення: точка зламу», міні-хакатонів «Зворотне проєктування». Запропоновано комплекс діагностичних методик для визначення рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Основні положення, результати та висновки дослідження можуть бути **використані** в системі професійної підготовки майбутніх фахівців з комп'ютерної інженерії, у системі післядипломної освіти, у самоосвітній діяльності майбутніх фахівців.

Результати дослідження **впроваджено** в освітній процес Державного закладу «Луганський національний університет імені Тараса Шевченка» (довідка № 1/522 від 09.06.2026 р.); Національного університету біоресурсів і природокористування України (довідка № 10 від 29.05.2026 р.); Державного університету «Київський авіаційний інститут» (довідка № 1.14/15д від 04.05.2026 р.).

Представлені в дисертаційній роботі наукові положення отримані автором самостійно. **Особистий внесок здобувача** у статті, опублікованій у співавторстві з В. Прилипко та І. Сохань, полягає в обґрунтуванні вибору інноваційних методик: метод проєктів і мозковий штурм як базових технологій проблемно зорієнтованого навчання; проведенні емпіричного оцінювання їх впровадження у підготовку фахівців технічних спеціальностей. У статті, опублікованій у співавторстві з В. Лахно С. Волошиным, С. Мамченко та М. Лахно, полягає в розробці програмної реалізації мережі Байєса мовою Python для прозорого аналізу DDoS-атак; адаптації створеного математичного інструментарію як практичного кейсу, що може використовуватися у технологіях проблемно зорієнтованого навчання майбутніх фахівців. У статті, опублікованій у співавторстві з Д. Касаткіним, С. Волошиным та Б. Гусевим, полягає у визначенні ефективності застосування розробленої системи підтримки прийняття рішень на основі порівняльного аналізу результатів експертного оцінювання рівня захищеності, у статті, опублікованій у співавторстві з В. Лахно та С. Мамченко, полягає у представленні гібридного методу виявлення кіберзагроз у мережевому трафіку університету. В опублікованому у співавторстві з О. Коваленко, С. Волошиным, Б. Гусевим та Є. Нікітенко навчальному посібнику «Комп'ютерна схемотехніка» (2023) особистий внесок автора полягає у підготовці розділу 2.1 щодо проєктування лічильників на базі окремих тригерів, а також розробці практичної складової (прикладів розв'язання задач, контрольних завдань та запитань для самоконтролю здобувачів освіти) до розділів 2.1, 2.2 та 3.1.

Апробація результатів дослідження. Основні теоретичні положення, результати та висновки дисертаційного дослідження обговорювалися на засіданнях кафедри педагогіки ДЗ «Луганський національний університет імені

Тараса Шевченка», репрезентовано на Всеукраїнській науково-практичній конференції «Актуальні питання документознавства: історія та сьогодення» (Полтава, 2023), Міжнародній науково-практичній конференції «Глобальні та регіональні проблеми інформатизації в суспільстві і природокористуванні'2024» (Київ, 2024), VIII Міжнародній науковій конференції «Інноваційні тенденції сьогодення в сфері природничих, гуманітарних та точних наук» (Тернопіль, 2026), XVIII Міжнародній науково-практичній конференції «Теорія і практика сучасної науки та освіти» (Львів, 2026), V Міжнародній науково-практичній конференції «Recent Trends in Science» (Дніпро, 2026), II Міжнародній науково-практичній конференції «Освіта в сучасних викликах» (Лубни, 2026), V Міжнародній науково-практичній конференції «Наука і освіта в глобальному та національному вимірах: виклики, загрози, перспективи розвитку» (Лубни, 2026). Результати дослідження обговорювалися на методологічному онлайн-семінарі молодих дослідників «Компетентнісні засади освітнього процесу в умовах впровадження інновацій: вітчизняний та міжнародний досвід» (Житомир-Полтава-Київ, 2023), міжуніверситетському семінарі «Практичний досвід постановки лабораторної роботи та її проведення: дистанційне та очне проведення занять» (Житомир, 2024).

Публікації. Основні положення й результати дисертаційної роботи висвітлено в 14 публікаціях, з яких: 4 статті у наукових фахових виданнях, з них 1 – у співавторстві; 9 публікацій апробаційного характеру, з них – 3 у співавторстві; навчальний посібник у співавторстві.

Структура й обсяг дисертації. Дисертаційна робота складається зі вступу, двох розділів, висновків до кожного розділу, загальних висновків, списку використаних джерел (223 найменування, з них – 55 іноземними мовами) та 4 додатків на 98 сторінках. Ілюстративний матеріал поданий у 20 таблицях і 14 рисунках. Загальний обсяг дисертації – 353 сторінки, із яких основного тексту – 232 сторінки.

РОЗДІЛ І
ТЕОРЕТИКО-МЕТОДИЧНІ ЗАСАДИ ФОРМУВАННЯ
МЕТОДОЛОГІЧНОЇ КОМПЕТЕНТНОСТІ МАЙБУТНІХ ФАХІВЦІВ З
КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ЗАСОБАМИ ТЕХНОЛОГІЙ
ПРОБЛЕМНО ЗОРІЄНТОВАНОГО НАВЧАННЯ

1.1. Поняття «методологічна компетентність» у сучасному соціогуманітарному дискурсі

В умовах динамічних змін та високої турбулентності ринку праці методологічна підготовка фахівців перетворюється із суто академічного елемента освіти на стратегічний інструмент професійного виживання та успіху. Стрімкий розвиток технологій, економічна нестабільність та регулярна поява нових викликів призводять до швидкого застарівання конкретних прикладних знань і навичок (так званих «*hard skills*»). У такому середовищі фахівець, чиє навчання обмежувалося лише засвоєнням готових алгоритмів, опиняється під загрозою професійної дезадаптації. Натомість фундаментальна методологічна підготовка озброює професіонала культурою мислення, яка дозволяє бачити системні зв'язки, самостійно відсіювати інформаційний шум і швидко опановувати нові предметні сфери відповідно до запитів часу.

Головна цінність методологічно підготовленого фахівця в часи невизначеності полягає в його здатності ефективно діяти за відсутності чітких інструкцій та готових шаблонів рішень. Ринок праці все частіше висуває вимоги до фахівців різного профілю – від інженерів та ІТ-спеціалістів до управлінців і гуманітаріїв – щодо вміння працювати в умовах дефіциту або суперечливості даних. Методологічна компетентність дає змогу не просто адаптуватися до змін, а проактивно конструювати нові знання: правильно ставити дослідницькі чи практичні завдання, розробляти унікальні стратегії дій та прогнозувати наслідки прийнятих рішень. Професіонал із розвиненим методологічним мисленням сприймає кризу не як глухий кут, а як простір для пошуку альтернативних, нестандартних підходів.

Високий рівень методологічної підготовки забезпечує фахівцю конкурентоспроможність та інваріантність на ринку праці, мінімізуючи ризики втрати актуальності через автоматизацію чи зникнення окремих професій. Методологічні знання є надпредметними, можуть екстраполюватися з однієї галузі в іншу, полегшуючи процеси міждисциплінарної взаємодії. Фахівець, який володіє методологічним інструментарієм, стає автономним суб'єктом власного професійного розвитку, здатним самостійно вибудовувати освітню траєкторію протягом усього життя.

Термінологічну базу дослідження методологічної компетентності фахівців різного профілю становить інтегрована система понять, що відображає міждисциплінарний характер цього феномену на перетині філософії науки, професійної педагогіки, епістемології. Ключовими конструктами виступають такі поняття, як «методологія», «компетентність», «компетентнісний підхід», «методологічна компетентність», а також такі дотичні до теми дослідження поняття, як «методологічна культура», «дослідницька компетентність».

Потлумачення поняття «методологія» у словникових виданнях представлено у Таблиці 1.1.

Таблиця 1.1.

Поняття «методологія» у словникових виданнях

Автор, видання	Визначення
Гончаренко С. Український педагогічний словник (1997) [25, с. 281]	вчення про методи пізнання та перетворення дійсності
Філософський енциклопедичний словник (2002) [157, с. 374]	1) сукупність підходів, способів, методів, прийомів та процедур, що застосовуються в процесі наукового пізнання та практичної діяльності для досягнення наперед визначеної мети. Такою метою в науковому пізнанні є отримання об'єктивного істинного наукового знання або побудова наукової теорії та її логічне обґрунтування, досягнення певного ефекту в експерименті чи спостереженні тощо. 2) галузь теоретичних знань і уявлень про сутність і форми, закони, порядок та умови застосування підходів, способів, методів, прийомів та процедур в процесі наукового пізнання та практичної діяльності

Merriam-Webster's Collegiate Dictionary 2003) [200, с. 782]	1а. Сукупність методів, правил і постулатів, що застосовуються певною науковою дисципліною; конкретна процедура або набір процедур. 1б. Аналіз принципів або процедур дослідження в певній галузі; 2. Систематичне вивчення методів, які застосовуються, можуть бути застосовані або застосовувалися в межах певної дисципліни.
Шапар В. Б. Сучасний тлумачний психологічний словник (2007) [164, с. 249]	система принципів і способів організації та побудови теоретичної і практичної діяльності, а також вчення про цю систему. Втілюється в організації та регуляції всіх видів людської діяльності
Енциклопедія освіти (2008) [33, с. 498]	учення про метод, наука про побудову людської діяльності. Традиційно найрозвиненішою галуззю методології є методологія пізнавальної діяльності, методологія науки
Великий тлумачний словник сучасної української мови (2009) [14, с. 664]	1. Вчення про науковий метод пізнання й перетворення світу; його філософська, теоретична основа. 2. Сукупність методів дослідження, що застосовується у будь-якій науці відповідно до специфіки об'єкта її пізнання.
Гладун А.Я. та ін. Англо-український словник термінів з інформаційних технологій та кібербезпеки (2018) [23, с. 227]	1. Наука про методи пізнання й перетворення світу; 2. Сукупність прийомів дослідження, що їх застосовують у будь-якій науці відповідно до специфіки об'єкту її пізнання
Національний освітньо-науковий глосарій (2018) [98, с. 50]	вчення про методи пізнання і перетворення світу. Сукупність прийомів дослідження, що їх застосовують у будь-якій науці відповідно до специфіки об'єкту її пізнання
Словник термінів ІТ і комп'ютерної інженерії (2025) [139, с.410]	Вчення про метод. Термін «методологія» позначає теорію цілеспрямованої пізнавальної діяльності. Завданням методологічних досліджень в рамках конкретної природничо-наукової області є формулювання тверджень, що визначають оптимальні способи досягнення цілей цієї науки. Рішення вказаної задачі може бути знайдене шляхом аналізу як цілей науки, так і специфіки її об'єктів. Принципи формування прийомів дослідження, вживаних у тій або іншій галузі науки. Сукупність методів, застосовуваних в життєвому циклі розробки ПЗ, об'єднаних одним загальним філософським підходом (Градів Буч). Система принципів, дій і процедур, вживаних в конкретній галузі знань.

Аналіз словникових дефініцій різних років та наукових галузей (педагогіки, філософії, психології, освіти та інформаційних технологій) дозволяє зробити узагальнюваний висновок про те, що поняття «методологія» традиційно розглядається у двох взаємопов'язаних аспектах: *теоретико-фундаментальний*, що розглядає це поняття як цілісне вчення про метод, теоретичну галузь знань та систему принципів, що досліджує закони, умови, форми й оптимальні способи організації та побудови як науково-пізнавальної, так і практичної діяльності людини; *практично-інструментальний*, що пов'язано із конкретною сукупністю підходів, методів, правил, процедур і прийомів, які систематично застосовуються

в межах певної предметної галузі для досягнення чітко визначеної мети та перетворення дійсності. Важливо підкреслити, що сучасні методологічні напрацювання в галузі методології акцентують увагу на постнекласичному розумінні наукового пізнання та ролі суб'єкта у такому пізнанні. Так. Л. Ваховський підкреслює, що «внутрішня етика науки, стимулююча пошук істини і орієнтацію на приріст нового знання, співвідноситься в цих умовах з загальногуманістичними принципами і цінностями. Наукове пізнання починає розглядатися в контексті соціальних умов його буття і його соціальних наслідків як особлива частина життя суспільства» [13, с. 160].

Відповідно, при розширенні сфери застосування поняття «методологія» (від класичного філософського й суто наукового пізнання до прикладних технологічних галузей, як-от ІТ-сфера), незмінним ядром методології залишається системність, цілеспрямованість та аналіз відповідності методів специфіці об'єкта.

У більшості науково-методичних посібників з методології наукових досліджень (як правило, для здобувачів другого та третього рівнів вищої освіти) наведено словникові визначення поняття «методологія». Як приклад, позначимо визначення з посібника «Основи наукових досліджень» для студентів спеціальності 123 «Комп'ютерна інженерія», де методологія розглядається, як «система принципів і способів організації та побудови теоретичної та практичної діяльності людини, а також вчення про таку систему» [107, с. 22-23].

У дослідженнях відомого українського методолога А. Фурмана поняття «методологія» увиразнено як «наука і мистецтво уможливлення досконалого мислення та ефективної діяльності в усіх сферах суспільного життя» [161, с. 27]. Важливим є зауваження про те, що «форми і методи, підходи і способи, процедури і техніки, засоби та інструменти проникають в усі сфери суспільно корисної практики, в буденні сегменти матеріального, інформаційного, технічного, соціального і духовного виробництва» [161, с. 30].

У наукових розвідках А. Фурмана репрезентовано концептуальну ідею щодо *методологічного мислення*, яке «фундується не на критерії істинності як на

відповідності знання об'єкту, а на критерії реалізованості як відповідності багатополісній, а саме об'єкта збігості проєкту, програмі чи сценарію, діяльності – припису, нормі, еталону, самого мислення – мислимому в ідеальній площині свідомості, що об'єктивоване як мислепродукт у тій чи іншій невербальній формі, рефлексії – змісту діяльності, мислєдіяльності, професійного методологування» [163, с. 219]. Відтак, саме професійне методологування має розглядатися як важливий аспект професійної підготовки майбутніх фахівців різного профілю.

Важливо підкреслити, що у контексті конкретної предметної галузі поняття «методологія» набуває свого специфічного значення. Так, у межах галузі А1 Освітні науки використовуються поняття «методологія освіти», «методологія педагогіки», які достатньо ґрунтовно репрезентовані у сучасних наукових дослідженнях.

Відзначимо, що у межах спеціальності 123 / F7 «Комп'ютерна інженерія» поняття «методологія», як правило, використовується для опису сукупності методів, принципів, підходів та інструментів, що застосовуються для проєктування, розробки, тестування та супроводження комп'ютерних систем та їхніх компонентів (як апаратного, так і програмного забезпечення). Відповідно, таке поняття деталізується через такі позиції: методології проєктування та розробки (наприклад, Agile, Scrum, DevOps), що пов'язано із визначенням життєвого циклу створення обчислювальних систем, організацією командної роботи та процесом управління змінами; методології архітектурного моделювання (підходи до побудови складної ієрархії взаємодії апаратних засобів (*Hardware*) та системного програмного забезпечення (*Software*); методології верифікації та тестування (системні підходи до перевірки надійності, відмовостійкості та продуктивності комп'ютерних мереж і систем.

На Рис. 1.1. представлено концепт-карту «Методологія комп'ютерної інженерії». Використання концепт-карт дозволяє унаочнити саме структурно-функційні характеристики об'єкта чи процесу завдяки відповідним дескрипторам-дієсловам.

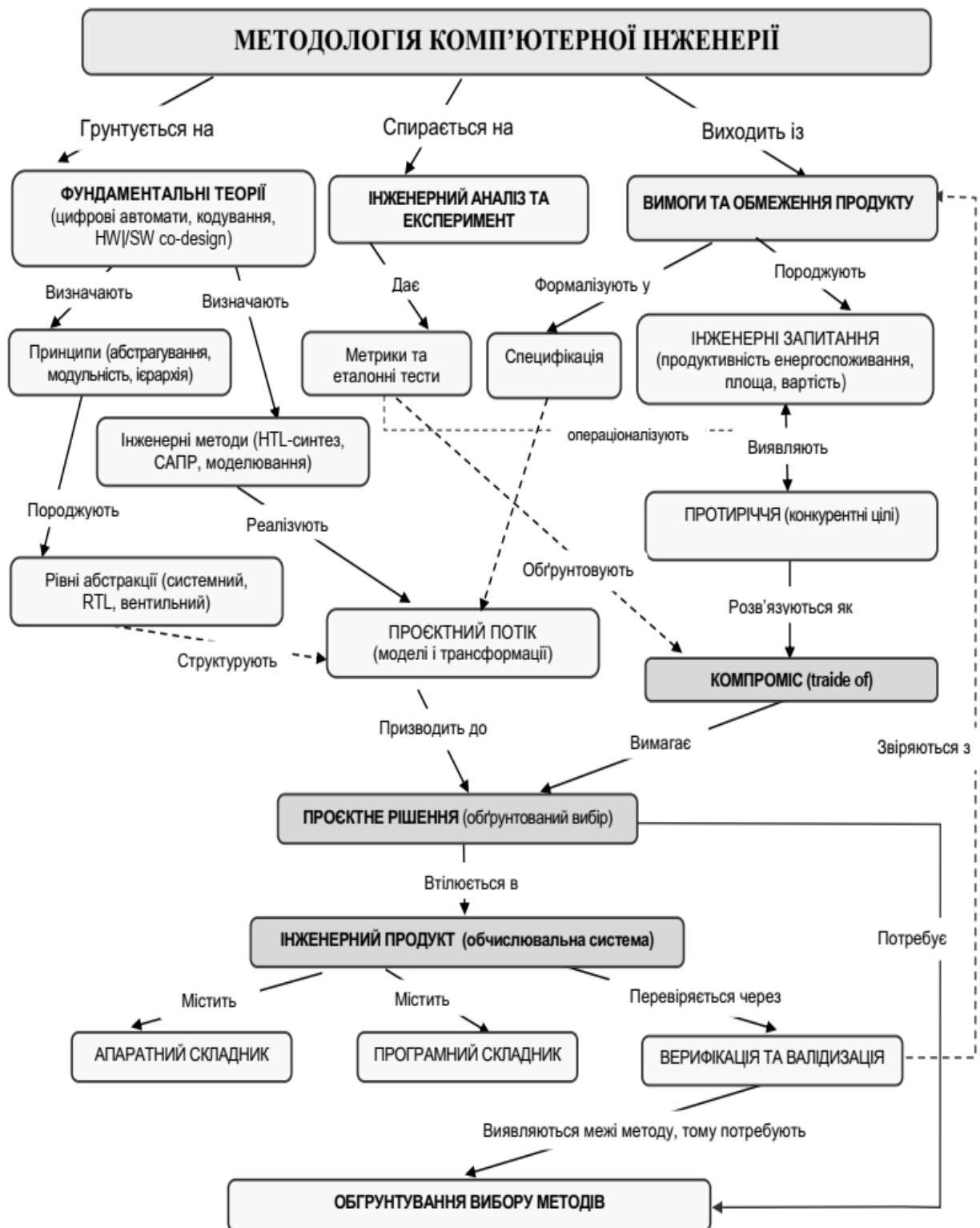


Рис. 1.1. Концепт-карта «Методологія комп'ютерної інженерії»

Необхідно підкреслити, що загальна методологія забезпечує фундаментально-системний базис пізнання та мислення у процесі професійної підготовки фахівців ІТ-сфери, тоді як методологія комп'ютерної інженерії

конкретизує ці універсальні принципи у сфері проєктування й розробки апаратно-програмних комплексів, утворюючи в їхній єдності цілісну основу для формування методологічної компетентності майбутнього фахівця.

Таким чином, значення поняття «методологія» є контекстуально залежним і трансформується відповідно до специфіки наукової чи практичної галузі – від фундаментально-теоретичного підґрунтя в освітніх науках до прикладного, системно-інженерного інструментарію в комп'ютерній інженерії.

У контексті позицій щодо методології важливе значення має поняття «метод», сутнісні ознаки якого представлено у Таблиці 1.2.

Таблиця 1.2.

Поняття «метод» у словникових та науково-методичних виданнях

Автор, видання	Визначення
Гончаренко С. Український педагогічний словник (1997) [25, с.205]	спосіб організації практичного та теоретичного освоєння дійсності, зумовлений закономірностями об'єкта, що розглядається
Філософський енциклопедичний словник (2002) [157, с. 373]	систематизований спосіб досягнення теоретичного чи практичного результату, розв'язання проблем чи одержання нової інформації на основі певних регулятивних принципів пізнання та дії, усвідомлення специфіки досліджуваної предметної галузі і законів функціонування її об'єктів
Шапар В. Б. Сучасний тлумачний психологічний словник (2007) [164, с. 248-249]	спосіб організації діяльності, що обґрунтовує нормативні засоби здійснення наукового дослідження. Шлях дослідження, що впливає з загальних теоретичних уявлень про сутність досліджуваного об'єкта
Енциклопедія освіти (2008) [33, с. 486, 487]	спосіб побудови й обґрунтування системи філософського знання; сукупність прийомів і операцій практичного й теоретичного освоєння дійсності... Науковий метод – це цілеспрямований підхід, шлях, за допомогою якого досягається поставлена мета. Це комплекс різних пізнавальних підходів і практичних операцій, спрямованих на набуття наукових знань.
Великий тлумачний словник сучасної української мови (2009) [14, с. 664]	1. Спосіб пізнання явищ природи та суспільного життя. 2. Прийом або система прийомів, що застосовується в якій-небудь галузі діяльності (науці, виробництві тощо).
Гончаренко С. У. Педагогічні дослідження: методологічні поради молодим науковцям (2010) [27, с.123]	спосіб пізнання реальності, що вивчається, який дає можливість розв'язати завдання і досягти мети пошукової діяльності
Сисоєва С. О., Кристопчук Т.Є. Методологія науково-педагогічних досліджень (2013) [131, с. 86]	спосіб пізнання явищ дійсності, їх взаємозв'язків і розвитку.

Гладун А. Я. та ін. Англо-український словник термінів з інформаційних технологій та кібербезпеки (2018) [23, с. 227]	1. Система принципів та способів пізнавально-теоретичної і практичної діяльності. 2. Спосіб, прийом або система прийомів для досягнення якої-небудь мети, для виконання певної операції
Основи наукових досліджень. Конспект лекцій: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія» (2022) [107, с. 23]	сукупність прийомів або операцій практичного чи теоретичного освоєння дійсності, підпорядкованих вирішенню конкретної задачі.

У контексті методологічної підготовки фахівців різного профілю важливим є твердження А. Фурмана про те, що «метод входить до структури будь-якої довершеної діяльності – від рефлексивних філософування й теоретизування до суто технічно-виконавської та побутової, й відтак є обов'язковим складником, що забезпечує внутрішню організацію й регуляцію її процесного перебігу за тих чи тих соціокультурних умов» [160, с. 49]. У дисертації Л. Бутенко наведено таку позицію філософа Л. Зеленова: «якщо теорія – це знання, що звернено до об'єкта, гносеологічне знання, то метод – це те ж знання, але звернене до суб'єкта, до діяча, до діяльності, до процесу отримання нового знання та його використання в діяльності людини» [Цит. за: 11, с. 225].

Відповідно, з урахуванням проблематики нашого дослідження важливо відзначити, що опанування *методів професійної діяльності* необхідне для формування у майбутніх фахівців з комп'ютерної інженерії цілісного, інтегративного, динамічного та системного стилю мислення. Це дозволяє їм бачити взаємозв'язки між різними рівнями комп'ютерних систем, що передбачає мережеву взаємодію освітніх компонентів на основі міждисциплінарності та конвергенції знань, що забезпечує глибоку інтеграцію теорії та практики в ІТ-галузі. У навчальному посібнику «Методика та оптимізація наукових досліджень» для здобувачів вищої освіти спеціальності 123 «Комп'ютерна інженерія» акцентовано увагу на необхідності комплексного використання загальнонаукових методів (аналіз, синтез, порівняння, узагальнення та ін.) та методів, що є специфічними для ІТ-галузі (апріорне та математичне моделювання, методи відображення та перетворення інформації в комп'ютерних системах та ін.) [125]. «Золотим стандартом» у світовій практиці підготовці

фахівців ІТ-галузі вважається підручник Д. Гарріса та С. Гарріс «Цифровий дизайн та архітектура комп'ютера» (*Digital Design and Computer Architecture*), який неодноразово перевидавався та репрезентує численні методи означеної галузі [189].

Таким чином, навчання методів професійної діяльності майбутніх фахівців в цілому та фахівців ІТ-сфери зокрема має відбуватися на основі єдності їхньої загальнометодологічної та фахової підготовки, принципів системного підходу, де опанування окремих методів реалізації професійних завдань підпорядковане формуванню методологічної компетентності майбутнього фахівця.

Існують різні підходи до класифікації методів наукового пізнання, оскільки в їх основу вчені покладають різні критерії залежно від цілей та специфіки дослідження. Найбільш поширеним є поділ методів за ступенем їхньої загальності та сферою застосування, де традиційно виокремлюють філософські, загальнонаукові та конкретно-наукові підходи. Традиційно у науковому пізнанні розрізняють теоретичні методи, призначені для виявлення закономірностей відповідного об'єкта або процесу, та емпіричні методи, що використовуються для безпосереднього дослідження фактів дійсності, систематизації даних, їхньої інтерпретації на основі певних теоретичних засад. Відповідно, у процесі методологічної підготовки майбутніх фахівців суттєве значення має ознайомлення із сутністю та алгоритмами використання різних методів наукового пізнання та реалізації завдань професійної діяльності.

Наукові розвідки з проблем методології, як правило, акцентують увагу на методологічних аспектах методології наукових досліджень. Однак, дослідники звертаються також до методології практичної діяльності. Так, у контексті професійної підготовки майбутніх учителів Л. Бутенко зауважує, що методологічні регулятиви – інструмент не лише теоретичної, але й практико перетворювальної діяльності [11, с. 170]. Показовим є твердження А. Фурмана про те, що якщо «метод входить до структури будь-якої довершеної діяльності – від рефлексивних філософування й теоретизування

до суто технічно-виконавської та побутової, й відтак є обов'язковим складником, що забезпечує внутрішню організацію й регуляцію її процесного перебігу за тих чи тих соціокультурних умов» [160, с. 49].

У контексті методологічної підготовки майбутніх фахівців різного профілю особливого значення набувають позиції дослідників щодо науково-дослідницького та практико зорієнтованого виміру методології. Так, М. Кубаєвський та А. Фурман зауважують, що «у першому випадку методологію розуміють як науку, що узагальнює і систематизує знання про методи наукового пізнання, тобто як філософську теорію методів, у другому – як застосування теорії до вирішення практичних завдань чи проблем, себто як самостійну прикладну дисципліну, що цілеспрямовано відрефлексована від формальної логіки і гносеології» [63, с. 47]. Показовим є твердження А. Фурмана про те, що «форми і методи, підходи і способи, процедури і техніки, засоби та інструменти проникають в усі сфери суспільно корисної практики, в буденні сегменти матеріального, інформаційного, технічного, соціального і духовного виробництва [161, с. 30].

Поняття «методологія практичної діяльності» увиразнено у дослідженні В. Гречаного та В. Сабадухі, які відзначають, що «потреба в методології в індивіда загострюється в приховано проблемних ситуаціях, коли він не може отримати результат, що задовольняє життєзабезпечення, діючи старими засобами, прийомами, способами, методами, коли він не знає, в чому полягає проблема, на які завдання її можна розкласти та якими новими засобами, прийомами, способами, методами, які ще належить якимось чином розробляти, розв'язати» [29, с. 104]. Відповідно, у сучасному швидко змінюваному світі актуальною є «відмова від розуміння методології виключно як надбудови над теоретичними засадами предметного знання та прийняття методології як сфери практичної діяльності, спрямованої на створення технологій трансформації, виробництва нового знання та інноваційних ідей з урахуванням специфіки методологічного мислення» [11, с. 193]. Підкреслимо також висновок про те, що «практичне знання у такому контексті пов'язано не тільки із змістом відповідної предметної

галузі, а й передбачає вироблення особистісного ставлення, теоретико-методологічної установки щодо форм та способів отримання та використання такого знання, рефлексивне ставлення до процесу пізнання» [11, с. 191].

Важливим складником термінологічного поля нашого дослідження є **поняття «компетентність»**. Концептуальні витoki обґрунтування компетентнісного підходу в освіті пов'язані із науковими положеннями Дж. Равена [205], який розглядав компетентність не як статичний набір ізольованих знань чи навичок, а як складну, динамічну систему вищих психологічних характеристик, що тісно пов'язані з внутрішньою мотивацією та особистісними цінностями людини. У межах його підходу стверджується, що ефективна діяльність можлива лише тоді, коли індивід виконує значущі для нього завдання, оскільки саме особистісний інтерес активує внутрішні ресурси, такі як готовність ризикувати, здатність аналізувати перешкоди чи вміння залучати інших до співпраці. Згідно з Дж. Равеном, компетентнісний підхід вимагає створення специфічних соціальних та організаційних умов, які б вивільняли ініціативу людей і дозволяли їм ефективно взаємодіяти з суспільними інститутами [205]. Таким чином, трансформується фокус із суто професійного інструментарію на загальну здатність людини діяти самостійно, критично мислити та брати на себе відповідальність у мінливому сучасному світі.

Закон України «Про вищу освіту» потрактовує компетентність як «динамічну комбінацію знань, умінь і навичок, способів мислення, професійних, світоглядних і громадянських якостей, морально-етичних цінностей, які визначають здатність успішно здійснювати професійну та навчальну діяльність і є результатом навчання на певному рівні вищої освіти» [35]. У Національному освітньому глосарії з вищої освіти (2024) поняття «компетентність» представлено, як «здатність особи успішно соціалізуватися, навчатися, провадити професійну діяльність, яка виникає на основі поєднання знань, умінь, навичок, способів мислення, поглядів, цінностей, інших особистих якостей», компетентнісний підхід – як «підхід до розроблення та реалізації освітніх програм (освітніх компонентів), що передбачає визначення та опис запланованих результатів

навчання, яких має досягти студент по завершенню навчання за освітньою програмою (освітнім компонентом) та які можуть бути виміряні й оцінені, спрямованих на розвиток відповідних визначених компетентностей» [97, с. 43].

Вітчизняна нормативно-правова база у галузі вищої освіти стосовно компетентнісного підходу суголосна документам Європейського освітнього простору. Так, у дослідженнях комісії Ради Європи сутність поняття «компетентність» визначається як уміння, кваліфікації, уявлення, або опорні знання, які є ключовими у професійній діяльності і використовуються у нових ситуаціях [196, с. 13-21], Європейська рамка кваліфікацій для навчання впродовж життя (до 2017 р.) визначає компетентність «як «підтверджену здатність використовувати знання, уміння, навички, а також особисті, соціальні та/або методологічні здібності в робочих чи навчальних ситуаціях, у професійному та особистісному розвитку» [97, с. 43]. Європейська рамка електронних компетенцій (*e-Competence Framework*) як еталонна основа концепцій, що застосовуються в секторі інформаційно-комунікаційних технологій (ІКТ) потрактовує компетентність як продемонстровану здатність застосовувати знання, навички та ставлення для досягнення помітних (вимірюваних) результатів [183, с. 5].

Концептуальні засади компетентнісного підходу в освіті в цілому та у вищій освіті зокрема представлено у наукових працях таких дослідників, як Н. Бібік, О. Овчарук, О. Пометун [58], В. Луговий [77], О. Жукова та А. Комишан [57] та ін.

О. Овчарук наголошує, що за результатами позицій експертів ЄС компетентність розглядається як «здатність застосовувати знання та вміння ефективно й творчо в міжособистісних відносинах – ситуаціях, що передбачають взаємодію з іншими людьми в соціальному контексті так само, як і в професійних ситуаціях. Компетентність – поняття, що логічно походить від ставлень до цінностей, та від умінь до знань» [58, с. 7]. Надзвичайно актуальним є твердження І. Беха про те, що «головною метою цього підходу має бути виховання людини як *суб'єкта практичної дії*. Світ нинішньої практики надзвичайно розмаїтий, і людина повинна в ній не лише орієнтуватися, а й

розумно поводитися, доцільно її змінюючи. Продуктивність практичної діяльності суб'єкта безпосередньо залежить від глибини знань, які вона мусить використовувати в цьому процесі» [2, с. 34].

Саме компетентності, як наголошував В. Луговий, виконують системоутворювальну роль у проектуванні освітніх програм майбутніх фахівців та організації освітнього процесу [77], оскільки акцент у навчання має трансформуватися з того, що знає, хоче і може викладач, на те, що потрібно тому, хто навчається. Відповідно, за умови компетентнісного підходу акцент зміщується з процесу на результат освіти. Важливо також відзначити, що компетентність розглядається також як комплекс *soft* і *hard skills*, тобто соціально-комунікативні та професійні навички [207], містить такі складники, як *know-how* (знати як діяти), де усвідомлення приходить лише через практику; *know-what* (знати що), тобто інтеграцію численних видів здібностей, навичок і знань; схильність до навчання, як-от особистісна характеристика, ставлення, умотивованість; уміння «практично й оперативно застосовувати знання в певних конкретних ситуаціях» [184, с. 45].

Компетентнісний підхід у вищій освіті – це «така організація навчального процесу, яка зосереджується на тому, що студенти в результаті навчання виконують чи вміють робити, а не на тому, чого вони мають навчатися. Навчання на основі компетентнісного підходу формує у студентів якості для реалізації професійної діяльності, які необхідні для ринку праці, а критерії та параметри оцінки результатів освіти уніфікуються і виражаються у термінах і результатах, які можуть бути інтерпретовані і враховані у будь-якому освітньому закладі будь-якої країни» [56, с. 14]. Сучасні науковці стверджують, що актуальність компетентнісного підходу у вищій освіті «обумовлюється необхідністю модернізації системи вищої освіти України, важливістю забезпечення умов для підготовки нової генерації критично мислячих і відповідальних фахівців, готових до постійного самовдосконалення, до життєдіяльності у сучасному суспільстві, ефективної професійної діяльності» [57, с. 254].

Таким чином, саме компетентнісний підхід на сьогодні є підґрунтям Стандартів вищої освіти підготовки здобувачів різного рівня, в тому числі й майбутніх фахівців з комп'ютерної інженерії [143].

Наукові положення щодо понять «методологія», «компетентність» зумовлюють розгляд поняття «методологічна компетентність» в контексті професійної підготовки майбутніх фахівців різного профілю (Таблиця 1.3).

Таблиця 1.3

**Наукові підходи до визначення поняття «методологічна компетентність»
фахівців різного профілю**

Автор	Визначення
М. Опачко (2009) [105, с. 275-276]	(<i>майбутнього вчителя фізики</i>) – система знань і умінь методологічного характеру, які становлять когнітивний та операційний компоненти компетентності
В. Кравцов (2010) [61, с. 114, 115]	(<i>майбутнього вчителя</i>) – комплексна готовність до теоретичної й практичної діяльності з перетворення педагогічної дійсності, заснована на методологічних знаннях; комплекс умінь по аналізу, синтезу, порівнянню, абстрагуванню, узагальненню, педагогічних феноменів, професійних важливих розумових якостей, таких як: аналогія, фантазія, гнучкість і критичність мислення
І. Козубцов (2014) [53, с. 160]	(<i>аспірантів системи вищої військової освіти</i>) – набута у процесі навчання інтегрована здатність аспірантів, що цілісно реалізується на практиці в процесі самостійної організації професійної творчої діяльності та складається з органічної єдності всіх її компонентів: методологічних знань, умінь, навичок з наукової, науково-педагогічної та військово-професійної діяльності, досвіду організації діяльності, цінностей і ставлення
Н. Кугай (2014) [64, с. 168]	(<i>майбутнього вчителя математики</i>) – потенційна властивість особистості, що виявляється в <i>готовності і здатності</i> майбутнього вчителя математики використовувати методологічні знання, вміння і навички для отримання нового наукового знання, для творчого осмислення власного досвіду, для проектування навчально-пізнавального процесу навчання математики в школі; наявність системи методологічних знань, умінь і навичок
О. Лаврентьєва (2014) [69, с. 239]	(<i>вчителя</i>) – сукупність методологічних знань, методологічних умінь і методологічних здібностей, що зумовлюють компетентне й професійне вирішення ним проблемних професійних завдань
О. Савицька (2019) [122, с. 73]	(<i>студентів</i>) – багатовимірне утворення, що містить методологічні знання в галузі наукової дисципліни; комплекс дослідницьких умінь (визначення причинно-наслідкових зв'язків; збереження логічних зв'язків з науковим арсеналом; проектування та конструювання дій та процесів; актуалізація та творче застосування теорії, концепції при пошуку розв'язку задачі),

	методологічну рефлексію, творчі здібності та мотиваційну складову
Лі Ліцюань (2020) [75, с. 50]	(<i>вчителя музичного мистецтва</i>) – системне особистісно-професійне, динамічне утворення, яке є здатністю вирішувати фахові завдання, що базується на мотиваційній, інформаційно-дослідницькій, рефлексивно-самовдосконалюваній спрямованості особистості; сформованості всезагальних, загальнонаукових, музично-педагогічних знань і вмінь та практичному виявленні їх у фаховій діяльності
Л. Нічуговська (2024) [100, с. 40]	(<i>аспірантів</i>) – термін «методологічна компетентність» як правило вживається для характеристики особи, яка вільно володіє й використовує методологічний апарат»
Л. Бутенко, С. Шехавцова (2026) [12, с. 2721-2722]	(<i>викладача вищої школи</i>) – складне інтегральне утворення, складник професійної компетентності викладача ЗВО, що визначає здатність педагога свідомо, системно та творчо застосовувати сукупність філософських, загальнонаукових і конкретно-наукових методів у науково-дослідній та освітній діяльності.

Важливою, на нашу думку, є позиція Н. Кугай, яка відносить методологічну компетентність до *ключових компетентностей* майбутніх учителів. Автор аналізує досвід педагогів Австрії, з точки яких «методологічна компетентність означає гнучкість, самоспрямоване навчання, здатність до незалежного вирішення проблем, самовизначення. Методологічна компетентність є вимогою для розвитку предметної компетентності – розуміння місця кожної науки у системі знань людства, «спосіб існування кожної науки» – розуміння діалектики отримання нових теоретичних знань та їх використання на практиці, незалежне оперування предметними знаннями та їх критичне осмислення з позицій практики та інших наук» [64].

Узагальнювальний аналіз наведених визначень свідчить про те, що методологічна компетентність є складним, багатоаспектним науковим феноменом, який дослідники трактують через різні, але взаємодоповнювальні родові категорії. Такими родовими категоріями постають:

- «інтегрована здатність» (І. Козубцов),
- «комплексна готовність» (В. Кравцов),
- «потенційна властивість особистості» (Н. Кугай),
- «системне особистісно-професійне утворення» (Лі Ліцюань),

- «складник професійної компетентності» (Л. Бутенко, С. Шехавцова).

Найчастіше автори визначають методологічну компетентність як набуту інтегровану здатність особистості, комплексну готовність до теоретичної й практичної діяльності або як системне особистісно-професійне і динамічне утворення. Водночас у науковому дискурсі зустрічаються підходи, де цю компетентність розглядають як потенційну властивість чи характеристику особи, яка вільно володіє методологічним апаратом, або ж як сукупність певних розумових якостей та комплекс мисленнєвих умінь.

Оцінюючи спільні риси у підходах науковців, можна стверджувати, що всі вони демонструють *єдність у розумінні практико-діяльній та дослідницькій природи цього поняття*. Фундаментом методологічної компетентності незмінно виступає система методологічних знань, умінь і навичок, яка слугує інструментом для розв'язання творчих, світоглядних, професійних та наукових завдань. Крім того, більшість дослідників виходять за межі суто знаннєвого підходу, включають до структури компетентності особистісно-мотиваційний та аксіологічний компоненти, зокрема цінності, стійкий інтерес до пізнання, критичність мислення та прагнення до наукового самовдосконалення.

Відмінності в наукових поглядах зумовлені насамперед специфікою професійної сфери, в контексті якої проводилися дослідження, а також вибором конкретних смислових акцентів. Так, І. Козубцов зосереджується на органічній єдності всіх компонентів компетентності, адаптуючи її до військово-професійної та науково-педагогічної діяльності аспірантів. В. Кравцов зміщує фокус на конкретні мисленнєві операції, такі як аналіз, синтез, аналогія чи фантазія, необхідні для перетворення педагогічної дійсності. Н. Кугай, спираючись на європейський досвід, наголошує на гнучкості та самоспрямованому навчанні, розглядаючи методологічну компетентність як передумову для успішного розвитку предметних знань. Універсальну трирівневу систему знань (від всезагальних до вузько-профільних) пропонує Лі Ліцюань, виокремлюючи чітку рефлексивно-самовдосконалювану спрямованість особистості у музично-педагогічній

сфері. На противагу таким розгорнутим структурам, Л. Нічуговська пропонує найбільш лаконічний, інструментальний підхід, пов'язуючи компетентність із безпосереднім вільним володінням методологічним апаратом.

Таким чином, попри різноманітність формулювань і специфічних акцентів, методологічна компетентність у сучасному науковому вимірі постає не просто як результат засвоєння теорії, а як цілісна, динамічна здатність фахівця творчо осмислювати дійсність, продукувати нове знання та ефективно проєктувати власну професійну діяльність.

Узагальнення наукових розвідок вітчизняних учених засвідчує, що обмежена кількість ґрунтовних наукових досліджень, присвячених проблематиці методологічної компетентності, зумовлена насамперед високим рівнем абстрактності та міждисциплінарної складності самого об'єкта аналізу. Методологічна компетентність у межах фундаментальних наукових галузей (філософія, педагогіки, психологія) та конкретного фахового змісту передбачає здатність до міждисциплінарної інтеграції знань, володіння комплексом методологічного інструментарію.

У цілому у сучасному соціогуманітарному дискурсі методологічна компетентність розглядається як динамічна інтегративна характеристика особистості фахівця, що передбачає опанування теорії та практики усвідомленого використання загальнонаукових та специфічних методів пізнання, орієнтованих на трансформацію дійсності на основі системного дослідження та проєктування професійної діяльності, ціннісного ставлення до обґрунтування та використання відповідного методологічного інструментарію, рефлексивної готовності до діяльності в умовах нестабільності ринку праці.

В англomовній науковій літературі поруч із поняттям «*Methodological Competence*» використовуються й інші поняття: *Research Literacy / Research Competence* (здатність застосовувати наукові методи), *Epistemic Competence* (епістемічна компетентність) як здатність індивіда розуміти природу наукового знання, межі власних епістемічних переконань, диференціювати наукові факти та припущення; *Critical Thinking and Reflective Practice*

(критичне мислення); *Systems Thinking* (системне мислення) як здатність бачити об'єкт цілісно; *Meta-cognitive Skills* (метакогнітивні навички) як знання про те, як ми пізнаємо світ.

У сучасних умовах розвитку суспільства формування методологічної компетентності перетворюється з додаткової переваги на засадничу вимогу професійної підготовки фахівців будь-якого профілю через глобальний перехід до постіндустріального суспільства та економіки знань. Необхідність такої підготовки зумовлена насамперед стрімким скороченням життєвого циклу технологій, за якого конкретні прикладні навички (*hard skills*) застарівають швидше, ніж триває цикл навчання у закладі вищої освіти. У таких умовах методологічна компетентність як здатність розуміти загальні принципи побудови знання та способи організації діяльності стає тим інваріантним ядром, що дозволяє фахівцеві не лише зберігати професійну придатність, а й самостійно конструювати нові компетенції відповідно до динамічних запитів реальності. Крім того, детермінантою виступає вимога до технологічної автономності та критичності мислення, що передбачає вміння фахівця самостійно обирати, верифікувати та комбінувати інструментарій для реалізації проєктів у цифровому середовищі. Ринок вимагає преадаптивності як здатності працювати на випередження, створюючи продукти, які будуть актуальними в умовах майбутніх стандартів, що неможливо без глибокого розуміння методологічних засад певної галузі. Таким чином, методологічна компетентність виступає як стратегічний ресурс особистості, що забезпечує її стійкість до інтелектуальної деградації та гарантує конкурентоспроможність у світі, де єдиною константою є постійні та непередбачувані зміни.

Для дослідження проблем методологічної компетентності фахівців різного профілю доцільно звернутися до таких дотичних понять, як **«методологічна культура»** та **«дослідницька компетентність»**.

У Таблиці 1.4. представлено підходи вітчизняних дослідників до трактування поняття «методологічна культура» фахівців різного профілю.

**Наукові підходи до визначення понять «методологічна культура»
фахівців різного профілю**

Автор	Визначення
В. Кравцов (2010) [61, с. 112-113]	<i>(майбутнього вчителя)</i> – цілісне багаторівневе і багатокомпонентне утворення у структурі його особистості, що містить знання педагогічної методології, володіння методами і методологією наукового пізнання, розумову діяльність в режимі методологічної рефлексії, розвинуті здібності критичного аналізу педагогічної дійсності, систему інтеріоризованих педагогічних та загальнолюдських цінностей, котрі визначають професійну свідомість, філософію педагогічної діяльності і детерміновані властивостями інтегральної індивідуальності
О. Лаврентьєва (2015) [70, с. 134, 135]	<i>(вчителя)</i> – у широкому значенні розглядається, як культура управління й конструювання, що містить у собі планування, організацію, мотивацію, контроль і творче перетворення професійно-педагогічної діяльності з урахуванням її динамічної та ймовірнісної структури, суб'єктів, цілей, внутрішніх і зовнішніх чинників; у вузькому – як культура мислення, специфічна «до сфери освіти, засновану на методологічних знаннях, уміннях, навичках, здатності до рефлексії, наукового обґрунтування, критичного осмислення й творчого застосування певних концепцій, форм і методів пізнання, управлінні й конструюванні педагогічної теорії і практики з метою вирішення професійно-педагогічних проблем, осмислення на раціогуманістичних засадах власного педагогічного досвіду, особистісно значущих чинників власного саморозвитку й розвитку особистісних структур свідомості своїх учнів
І. Сафонова (2016) [124, с. 188]	<i>(педагога-дослідника)</i> – здатність глибоко вникати в сутність педагогічних процесів, визначати провідні суперечності розвитку сучасної освіти, осмислювати і проєктувати стратегії методологічного пошуку через призму «людського в людині», творчо застосовувати сучасні концепції, підходи та методи пізнання, не порушуючи гуманітарної природи людини
Н. Лазаренко (2016) [71, с. 116]	<i>(викладача вищої школи)</i> – особлива культура наукового мислення, базою якої є методологічний світогляд, знання, вміння та навички, потрібні для виконання професійної діяльності, і досвід рефлексії (самопізнання та самооцінювання професіонала, який прагне вдосконалити не лише власну наукову чи педагогічну діяльність, а й систему вищої освіти загалом)
Л. Ковальчук (2025) [50, с. 232]	<i>(педагога-дослідника)</i> – у вузькому розумінні – як складова наукової культури педагога-дослідника, яка характеризує його здатність виконувати науково-дослідну діяльність з урахуванням різних рівнів методології і принципів наукового пізнання, використовувати методологічні підходи релевантно темі, предмету, меті та завданням наукового дослідження», у широкому – як інтегрована сукупність його методологічної компетентності, професійних і науково-дослідницьких якостей, у якій органічно поєднуються глибокі методологічні знання, методологічна рефлексія, дослідницькі вміння й навички, досвід творчої науково-дослідної та професійно-педагогічної діяльності, здатність до наукового пізнання на теоретичному й емпіричному рівнях, грамотного обґрунтування наукового апарату, теоретико-

	методологічних засад дослідження з використанням ієрархічної системи методології, методологічних підходів і принципів
О. Біліченко, О. Семенов (2025) [6, с. 17]	інтегральна якість особистості, що поєднує методологічну компетентність, дослідницькі навички й етичні принципи, сформовані у процесі академічної співпраці
Я. Топольник (2026) [152, с. 1082]	(викладача вищої школи) – комплексне професійно-особистісне утворення, яке виявляється у здатності інтегрувати глибокі теоретичні знання із гнучкою практичною діяльністю в динамічному освітньому середовищі.

Аналіз наведених підходів до визначення сутності поняття «методологічна культура» зумовлює висновок, що цей феномен розглядається не просто як сума професійних знань та умінь, а як складне, багаторівневе й системне утворення. Відзначимо такі *спільні для науковців положення*:

- розуміння методологічної культури як інтегративної якості, що обов'язково поєднує в собі глибокі теоретичні знання, розвинене наукове мислення та здатність до постійної професійної рефлексії;
- динамічний характер такої культури фахівців, що виявляється в активній творчій діяльності, здатності критично аналізувати й трансформувати навколишню дійсність, а також самостійно розв'язувати складні фахові проблеми;
- акцент на ціннісному вимірі, зв'язку професійної свідомості фахівців із загальнолюдськими орієнтирами та повагою до особистості;
- тенденція до дворівневого структурування методологічної культури, тобто виокремлення вузького спектру дослідницьких, інструментальних умінь та навичок та широкого контексту, пов'язаного із філософськими засадами навчальної та професійної діяльності.

Разом з тим, необхідно підкреслити, що відмінності між науковим осмисленням поняття «методологічна культура» пов'язані із специфікою професійного контексту, відповідними теоретичними акцентами. Так, відповідно до завдань педагогічної діяльності вчителя В. Кравцов та О. Лаврентьєва наголошують на значенні безпосередньої практичної, управлінської діяльності, де методологія слугує інструментом щоденного моделювання уроків, контролю, мотивації учнів та розвитку їхніх особистісних структур свідомості. Натомість,

Л. Ковальчук та І. Сафонова зміщують фокус на методологічну культуру педагога-дослідника, де на перший план виходить володіння саме академічним інструментарієм (обґрунтування наукового апарату, диференціація теоретичного й емпіричного рівнів пізнання, усвідомлення суперечностей розвитку освіти та проєктування стратегій наукового пошуку). Я. Топольник підкреслює значення епістемологічних засад методологічної культури викладача ЗВО, «осмислення природи, меж, джерел та методів отримання того знання, яке вони не лише продукують як науковці, а й транслують у студентську аудиторію» [152, с.1084].

Таким чином, відмінності полягають у масштабі застосування методологічного інструментарію – від конструювання конкретних практичних дій і розв’язання локальних професійних завдань до глобального наукового осмислення та реформування освітніх систем.

Важливим питанням у контексті нашого дослідження є взаємозв’язок методологічної компетентності та методологічної культури фахівців різного профілю. Такий взаємозв’язок науковці розглядають по-різному. Так, Л. Нічуговська відзначає, що «у багатьох наукових дослідженнях методологічна компетентність розглядається як незмінна складова методологічної культури аспіранта, що спрямована не лише на теоретичну діяльність, а й на організацію її практичної частини» [100, с. 40]. На зв’язок методологічної компетентності та методологічної культури в контексті підготовки майбутніх учителів та фахівців гуманітарних спеціальностей вказують О. Лаврентьєва та О. Малихін [70; 82]. В. Кравцов розглядає методологічну культуру як показник та передумову формування методологічної компетентності вчителя [61, с. 113], О. Біличенко та О. Семенов – як складник методологічної компетентності [6, с. 17].

Таким чином, співвідношення понять «методологічна компетентність» та «методологічна культура» дозволяє розглядати його через призму філософського зв’язку між інструментальним умінням людини та її внутрішнім світоглядом. Попри те, що обидва терміни описують здатність фахівця чи науковця ефективно організовувати свою діяльність, вони відображають різні рівні професійного та особистісного розвитку. Методологічна компетентність є

більш прагматичним, діяльнісним та інструментальним поняттям, оскільки визначає готовність і здатність людини застосовувати конкретні знання, методи, технології та алгоритми для вирішення практичних чи дослідницьких завдань. Натомість методологічна культура виходить за межі суто технічного вміння, інтегрує в собі не лише знання та навички, а й ціннісні орієнтації, етичні переконання, творче мислення та рівень усвідомлення відповідальності за результати своєї праці. Отже, ці поняття співвідносяться як частина і ціле. Методологічна компетентність виступає підґрунтям для формування методологічної культури. Водночас методологічна культура є вищим етапом розвитку особистості, де професійні навички стають частиною загального світогляду, трансформуючи звичайне виконання алгоритмів у глибоко усвідомлену, творчу та етичну діяльність.

Дотичним до поняття «методологічна компетентність» також слід вважати **поняття «дослідницька компетентність»**. У Таблиці 1.5 представлено наукові підходи до означеного поняття у контексті підготовки фахівців різного профілю.

Таблиця 1.5

Наукові підходи до визначення поняття «дослідницька компетентність»

Автор	Визначення
Л. Бондаренко (2015) [8, с.52]	(викладача ЗВО) – цілісна, інтегративна властивість особистості, що поєднує в собі знання, уміння, науковий досвід, особистісні цінності та якості (креативність, творче мислення, уяву, спостережливість) дослідника й виявляється в готовності ефективно здійснювати власну дослідницьку діяльність та здатності організувати продуктивну науково-дослідну роботу студентів.
С. Сисоєва, Л. Козак (2016) [132, с. 10-11]	інтегрована особистісно-професійна якість фахівця, яка відображає мотивацію до наукового пошуку, рівень володіння методологією педагогічного дослідження, особистісно значущими якостями дослідника, зокрема такими як інноваційне мислення, здатність до творчої та інноваційної діяльності
М. Вінник (2016) [17, с. 61]	(майбутніх інженерів-програмістів) – динамічна особистісна характеристика студента, що відображає прагнення та здатність (готовність) реалізувати свої знання, уміння, досвід, особисті якості для здійснення наукового дослідження в програмній інженерії та становить комбінацію мотивації й пізнавальних цінностей; інтегративних фахових та методологічних знань; багатофункціональних науково-дослідницьких умінь і рефлексії
М. Сорокопуд (2016) [142, с.88]	здатність реалізувати на практиці свій потенціал для успішної продуктивної діяльності в усіх сферах. Для майбутніх інженерів-програмістів – наявність міцних та глибоких знань з фізики,

	оволодіння методами та методикою дослідження, що необхідні для розв'язування задач дослідження фізичних явищ та процесів; сформовані важливіші дослідницькі вміння: виявляти проблему, формувати мету, задачі та гіпотезу, планувати проведення спостережень та експериментів, аналізувати вихідні дані та оцінювати результати дослідження; в науково-дослідницькій діяльності досягаються високі результати.
О. Біда (2017) [5, с. 11]	інтегративна характеристика особистості, яка передбачає володіння методологічними знаннями, технологією дослідницької діяльності, а також визнання їх цінності і готовність до їх використання у професійній діяльності.
С. Сапожников (2019) [123, с. 113]	інтегрована якість (характеристика) особистості, яка визначає її готовність і здатність до поетапного розв'язання дослідницьких задач, що виникають у реальних життєвих ситуаціях, у різних сферах діяльності, на основі використання знань та умінь із навчального й життєвого досвіду відповідно до сформованої системи загальнокультурних та професійних цінностей із метою отримання нових знань шляхом застосування методів наукового пізнання
І. Олійник (2022) [104, с. 310]	інтегративна характеристика особистості, що визначає здатність вирішувати наукові проблеми з урахуванням набутих теоретико-методологічних знань в конкретній науковій сфері, професійного та життєвого досвіду, цінностей та інтересів особистості та виявляється у спроможності та готовності самостійно освоювати та отримувати системи нових знань шляхом перенесення змістового контексту діяльності від функціонального до перетворювального, базуючись на наявних знаннях, уміннях, навичках та способах діяльності
М. Леврінц, Н. Варга, Ю. Попик (2023) [72]	інтегральна якість особистості, яка передбачає володіння вміннями і способами дослідницької діяльності з метою пошуку знань для вирішення освітніх проблем і побудови процесу навчання

З урахуванням основного фокусу нашої дисертаційної роботи ми навели лише деякі позиції сучасних вітчизняних науковців щодо поняття «дослідницька компетентність». Означене поняття доволі широко представлено у сучасному науковому дискурсі (наукові розвідки К. Василенко, С. Кравченко, В. Логвіненко, П. Лузана, С. Саяпіної, І. Солошич, В. Стрельнікова, О. Тітової, І. Щербань та ін.).

Серед наведених досліджень акцентуємо увагу на дисертації М. Вінника, який розглядає науково-дослідницьку компетентність майбутніх інженерів-програмістів, що є дотичним до проблематики нашої дисертаційної роботи. Автор підкреслює значення таких особистісних якостей майбутнього інженера-програміста: «професійні знання, професійний досвід, потреба у пізнанні, самоорганізація, самовдосконалення, творче мислення, уміння та

бажання навчатися, практичність, наукова культура, мотивація, лідерські якості, надійність, організаторські здібності, комунікабельність» [17, с. 62]. Розглядаючи дослідницьку компетентність майбутніх бакалаврів з комп'ютерної інженерії М. Сорокопуд відзначає необхідність якісної фундаментальної підготовки, навчання студентів працювати самостійно, здобувати знання з різних джерел інформації [142, с. 89].

Аналіз наведених позицій щодо поняття «дослідницька компетентність» зумовлює доцільність її розгляду в контексті проблем методологічної компетентності майбутніх фахівців, оскільки така якість перестає бути прерогативою виключно наукової сфери та перетворюється на базову вимогу ринку праці, оскільки озброює молодого спеціаліста інструментами для самостійного пошуку, критичного аналізу та структурування інформації. Здатність чітко формулювати проблему, висувати гіпотези, прогнозувати ризики та знаходити нестандартні рішення дозволяє випускникам успішно адаптуватися до професійної невизначеності, ефективно вирішувати складні практичні завдання та постійно оновлювати власні знання. Урахування висновків учених щодо сутності дослідницької компетентності зумовлює висновок про необхідність забезпечення етапності залучення майбутніх фахівців до наукового пошуку, урахування аксіологічного складника з метою формування методологічної компетентності як цілісного, професійно зорієнтованого особистісного утворення. Підкреслимо також значення і дослідницької, і методологічної компетентності для формування професійної компетентності майбутніх фахівців, необхідність їхнього включення у процесі навчання у ЗВО до квазіпрофесійної діяльності, що стимулюватиме розгляд тих чи інших методологічних питань не як абстракції, а як засобу перетворення професійної дійсності.

Таким чином, аналіз сучасного наукового дискурсу засвідчує, що поняття «методологічна компетентність» еволюціонувало від вузького розуміння як сукупності знань про методи дослідження до складного інтегрального утворення особистості фахівця. У контексті сучасних освітніх і професійних парадигм вона розглядається як здатність не просто застосовувати готові алгоритми, а глибоко

осмислювати, критично оцінювати та конструювати пізнавальну й практичну діяльність в умовах високої невизначеності. Науковці одностайно пов'язують цей феномен із розвитком системного мислення, рефлексивної культури та готовності до неперервного саморозвитку. Отже, методологічна компетентність виступає фундаментальним внутрішнім регулятивом, який визначає стратегію професійної поведінки фахівця, забезпечує його свідому методологічну позицію та інтегрує теоретичні знання у гнучкі інструменти для вирішення нестандартних завдань і генерації інновацій.

1.2. Сутність та структура методологічної компетентності майбутніх фахівців з комп'ютерної інженерії

Сутність та структура методологічної компетентності майбутніх фахівців з комп'ютерної інженерії має визначатися з урахуванням специфіки професійної діяльності таких фахівців, якісних ознак їхньої професійної компетентності та, відповідно, особливостей професійної підготовки.

Теоретико-методичні аспекти професійної діяльності майбутніх фахівців з комп'ютерної інженерії представлено в наукових працях таких вітчизняних дослідників, як О. Баранюк, М. Бирка, М. Вінник, О. Галицький, В. Говорун, І. Гомілко, М.-О. Єршов, В. Концедайло, Л. Макаренко, Н. Морзе, А. Пашков, М. Семенов, С. Семеріков, Л. Слободянюк, М. Сорокопуд, К. Стрюк, Ю. Шпильовий та ін., зарубіжних науковців – Й. Бернхард (J. Bernhard), М. Боррего (M. Borrego), П. Ванкат (P. Wankat), М. Гергардт (M. Gerhardt), Г. Мадхаван (G. Madhavan), М. Робінсон (M. Robinson), К. Сміт (K. Smith), Б. Фолкнер (B. Faulkner), Дж. Фройд (J. Froyd) та ін.

У статті Н. Морзе та О. Буйницької «Формування компетентностей майбутніх фахівців з програмної інженерії» (2015) зазначено перелік низки актуальних компетентностей таких фахівців. Наведемо ті компетентності, які отримали найвищі бали за результатами опитування роботодавців: «Аналізувати, теоретично і експериментально досліджувати методи, алгоритми, програми.

Створювати проєкти реінжинірингу технічних і бізнес-процесів, аналізувати результати. Використовувати сучасні засоби автоматизації проєктування для вирішення завдань підприємства. Виконувати системний аналіз вбудованих систем і семантичний опис даних і знань. Аналізувати організаційне оточення, існуючі системи, синтезувати вимоги до системи, вибирати ОМ, обґрунтовувати рішення» [96, с. 17-18]. Також наголошено на значущості таких компетентностей, як «вміння працювати в команді, вміння вирішувати проблеми, здатність до інновацій, практико-орієнтовані знання, знання суміжних наук і технологій, залежно від того, куди вбудовується система, висока якість виконання» [96, с. 18].

Професійну компетентність фахівця із комп'ютерної інженерії Л. Макаренко та Ю. Шпильовий визначають, як багаторівневе та динамічне утворення, «що об'єднує не лише глибокі знання та практичні навички, але й інтелектуальні, особистісні та соціальні якості, необхідні для успішної та ефективної професійної діяльності. Саме ця інтегрована якість дає можливість інженеру ефективно виконувати професійні завдання, розв'язувати складні технічні задачі та завдання, проєктувати, розробляти, впроваджувати та підтримувати комп'ютерні системи, пристрої та програмне забезпечення відповідно до сучасних вимог та стандартів» [80, с. 81]. Показовим є висновок дослідників про те, що така компетентність «не статичний набір характеристик, а динамічна сукупність знань, вмінь, навичок та особистісних якостей, що постійно розвиваються під впливом технічного прогресу та професійного досвіду. Вона є ключовим фактором конкурентоспроможності фахівця на ринку праці та його здатності робити значущий внесок у розвиток інформаційних систем і цифрових технологій» [80, с. 81].

У дисертації К. Стрюка наголошено, що професійна діяльність фахівців з комп'ютерної інженерії «передбачає виконання виробничих функцій: дослідницької (розробка та дослідження електронних схем, типових вузлів і пристроїв технічного обслуговування), проєктувальної (розробка прикладних програм, архітектури комп'ютера, системних програм, комп'ютерних мереж, робота з

базами даних), технічної (організація та обслуговування обчислювальних процесів, управління проєктами та ін.) та розв'язання різного класу завдань діяльності (стереотипних, діагностично-пошукових, евристичних)» [145, с. 2]. Відповідно, «професійна компетентність майбутнього фахівця з комп'ютерної інженерії» представлена як інтегральна характеристика особистості майбутнього фахівця, «яка поєднує знання, уміння, навички в галузі системної та апаратної інженерії й професійно важливі якості, що в сукупності забезпечують ефективне вирішення завдань професійної діяльності з обслуговування комп'ютерних систем і мереж» [145, с. 2-3]. Автор наголошує на значущості дослідницької функції, яка пов'язана із тим, що фахівці відповідної спеціальності повинні вміти «розв'язувати математичні та фізичні задачі шляхом створення відповідних застосувань, розв'язувати задачі теорії ймовірності і математичної статистики шляхом виконання відповідних перетворень, формулювати практичні задачі комп'ютерної логіки в термінах алгебри перемикальних функцій, абстрактної та структурної теорії цифрових автоматів, розрізняти і класифікувати проблеми фізичної реалізації інформаційних процесів в електронних приборах» [144, с. 19]. Дослідник також акцентує увагу на формуванні здатності до роботи в команді, до аналізу та синтезу науково-технічної природничо-наукової та загальнонаукової інформації [144, с. 121].

Вітчизняні дослідники також аналізують проблеми підготовки фахівців ІТ-сфери різних спеціальностей у зарубіжних університетах. Так, у дисертації Я. Сікори представлено детальний аналіз освітніх програм провідних світових університетів, порівняння міжнародних освітніх стандартів ІТ-галузі з підготовкою відповідних фахівців в університетах України [136, с. 80-115]. Заслужують на увагу позиції М. Якімової та О. Кін щодо підготовки бакалаврів у галузі електронної інформації в університетах КНР [167], які підкреслюють значущість індивідуалізації навчання, яка пов'язана із моделлю «набір за широким профілем з відкладеним вибором спеціальності», яка забезпечує поетапне й усвідомлене професійне самовизначення; надання можливості опанування поглиблених версій базових обов'язкових професійних курсів для

отримання фундаментальної фізико-математичної підготовки; наявність широкого спектру, об'єднаних у тематичні групи, вибіркових професійних курсів, що дозволяє конструювати індивідуальні комбінації дисциплін та формувати індивідуальний професійний профіль; запровадження гнучкого терміну навчання з урахуванням індивідуальних потреб та можливостей студентів; створення умов для академічного зростання здобувачів через міжнародну мобільність і побудову в майбутньому індивідуальної освітньо-кар'єрної траєкторії» [167, с. 1691].

У сучасних умовах, коли життєвий цикл конкретних технологій, мов програмування та програмних платформ є значно коротшим за період професійної підготовки фахівця, саме методологічна компетентність постає фундаментальним базисом, що забезпечує здатність до системного аналізу, стратегічного планування та швидкої адаптації до нових парадигм цифрової індустрії. За таких обставин методологічна компетентність у контексті комп'ютерної інженерії виступає як професійно зорієнтоване підґрунтя для усвідомленого вибору методів та методик реалізації фахових завдань, моделювання та оптимізації програмного забезпечення для різних сфер життєдіяльності суспільства. Це набуває особливої ваги у зв'язку з інтеграцією штучного інтелекту, розвитком кіберфізичних систем та хмарних обчислень, де критичне мислення та володіння загальнонауковими методами пізнання стають вирішальними для гарантування надійності та безпеки інженерних рішень.

Професійна підготовка майбутніх фахівців з комп'ютерної інженерії в умовах швидкої зміни технологічних парадигм вимагає відмови від вузькофункційної підготовки на користь методологічної універсальності. Студенти мають опанувати не лише конкретні мови опису апаратури, а й методологію системного проєктування, де ресурси (CPU, GPU, FPGA) розподіляються динамічно, виходячи з критеріїв архітектурної доцільності. Іншою суттєвою зміною є перехід від пріоритету обчислювальної потужності до парадигми енергоефективності та обмеженості ресурсів. Відповідно, актуальною постає необхідність формування в майбутніх інженерів «екологічного»

технічного мислення, спрямованого на багатокритеріальну оптимізацію систем. Паралельно з цим спостерігається наскрізна віртуалізація апаратних функцій у програмній оболонки (парадигма *Software-Defined Everything*), що стирає жорсткі межі між курсами архітектури комп'ютерів та системного програмування. Це вимагає від фахівця високого рівня абстрактного мислення та здатності оперувати віртуалізованими об'єктами як реальними фізичними структурами. Таким чином, швидка зміна парадигм у комп'ютерній інженерії перетворює здатність до самостійної концептуалізації нових знань та критичного відбору методів проектування на вагомий чинник професійної стійкості випускника в умовах постійних технологічних змін.

Зазначені позиції відображаються у нормативних документах та наукових дослідженнях питань підготовки інженерів в цілому та фахівців з комп'ютерної інженерії зокрема. Орієнтація на методологічний складник професійної підготовки відображена у Стандарті вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти [143] (Таблиця 1.6).

Таблиця 1.6

Відображення методологічного складника в Стандарті вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти

Інтегральна компетентність	Здатність розв'язувати складні спеціалізовані задачі та практичні проблеми під час професійної діяльності в комп'ютерній галузі або навчання, що передбачає застосування теорій та методів комп'ютерної інженерії і характеризується комплексністю та невизначеністю умов.
Загальні компетентності	Z1. Здатність до абстрактного мислення, аналізу і синтезу. Z2. Здатність вчитися і оволодівати сучасними знаннями. Z3. Здатність застосовувати знання у практичних ситуаціях. Z4. Здатність спілкуватися державною мовою як усно, так і письмово. Z5. Здатність спілкуватися іноземною мовою. Z6. Навички міжособистісної взаємодії. Z7. Вміння виявляти, ставити та вирішувати проблеми. Z8. Здатність працювати в команді.
Спеціальні (фахові, предметні) компетентності	P2. Здатність використовувати сучасні методи і мови програмування для розроблення алгоритмічного та програмного забезпечення. P3. Здатність створювати системне та прикладне програмне забезпечення комп'ютерних систем та мереж. P6. Здатність проектувати, впроваджувати та обслуговувати комп'ютерні

	системи та мережі різного виду та призначення. P12. Здатність ідентифікувати, класифікувати та описувати роботу програмно-технічних засобів, комп'ютерних та кіберфізичних систем, мереж та їхніх компонентів шляхом використання аналітичних методів і методів моделювання. P13. Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій. P15. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати та захищати прийняті рішення.
Результати навчання	N1. Знати і розуміти наукові положення, що лежать в основі функціонування комп'ютерних засобів, систем та мереж. N2. Мати навички проведення експериментів, збирання даних та моделювання в комп'ютерних системах. N6. Вміти застосовувати знання для ідентифікації, формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей. N7. Вміти розв'язувати задачі аналізу та синтезу засобів, характерних для спеціальності. N8. Вміти системно мислити та застосовувати творчі здібності до формування нових ідей. N11. Вміти здійснювати пошук інформації в різних джерелах для розв'язання задач комп'ютерної інженерії. N12. Вміти ефективно працювати як індивідуально, так і у складі команди. N13. Вміти ідентифікувати, класифікувати та описувати роботу комп'ютерних систем та їх компонентів. N14. Вміти поєднувати теорію і практику, а також приймати рішення та виробляти стратегію діяльності для вирішення завдань спеціальності з урахуванням загальнолюдських цінностей, суспільних, державних та виробничих інтересів. N15. Вміти виконувати експериментальні дослідження за професійною тематикою. N16. Вміти оцінювати отримані результати та аргументовано захищати прийняті рішення. N18. Використовувати інформаційні технології та для ефективного спілкування на професійному та соціальному рівнях. N19. Здатність адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення. N20. Усвідомлювати необхідність навчання впродовж усього життя з метою поглиблення набутих та здобуття нових фахових знань, удосконалення креативного мислення.

Отже, Стандарт вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти створює підґрунтя для реалізації завдань методологічної підготовки фахівців зазначеного профілю, фіксує увагу на необхідності розвитку системного та критичного мислення, здатності до

аналізу, синтезу, узагальнення та класифікації, оптимального вибору методів професійної та наукової діяльності. Загальна спрямованість нормативних документів у галузі підготовки ІТ-фахівців підтверджується вітчизняними та зарубіжними дослідниками. Так, серед тенденцій розвитку ІТ-освіти в Україні М.-О. Єршов зазначає орієнтацію на розвиток критичного та креативного мислення [34, с.148]. У межах структурно-функційного підходу до формування професійної компетентності бакалаврів з комп'ютерної інженерії Л. Слободянюк та М. Чумак наголошують на значенні проєктного навчання, яке сприяє розвитку аналітичного мислення та навичок командної роботи [137, с. 104].

Значущість системного мислення для інженера відзначає Г. Мадхаван (G. Madhavan), який стверджує, що «інженерне мислення» реалізується через універсальну систему вирішення проблем, що ґрунтується на трьох позиціях – структура (вміння бачити системи та зв'язки там, де інші бачать хаос), обмеження (робота в межах реальних ресурсів), компроміси (розуміння того, що будь-яке рішення має свою ціну) [198]. Відомі дослідники інженерної освіти М. Боррего (M. Borrego) та Й. Бернхард (J. Bernhard) підкреслюють, що методологічна підготовка є «окремим полем наукового пошуку», яке вимагає від студента розуміння філософії інженерної справи [174].

Професійна підготовка майбутніх фахівців з комп'ютерної інженерії у вітчизняних ЗВО здійснюється з урахування вимог низки міжнародних стандартів, зокрема таких, як *Computing Curricula 2020 (CC2020): Paradigms for Global Computing Education* [180], *Subject Benchmark Statement for Computing* (2022), що зумовлює перелік стандартів для випускників ІТ-програм від Агенції з забезпечення якості вищої освіти Великобританії (QAA) [213]. Відзначимо позиції *Computing Curricula 2020*, які суголосні проблемам формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії: здатність аналізувати, проєктувати та випробовувати цифрові логічні схеми, комбінаційні та послідовні автомати; уміння застосовувати методи формальної верифікації, логічні аналізатори та симулятори для діагностики й налагодження систем; гнучкість у сприйнятті нових технологій та інструментів; ефективна

робота в мультидисциплінарних командах; здатність вирішувати проблеми, незважаючи на труднощі; швидка реакція на зміни у вимогах чи робочому середовищі [180].

Серед професійних компетентностей ІТ-фахівців за документом «*Subject Benchmark Statement for Computing*» у межах нашого дослідження важливі такі: здатність створювати, аналізувати та оцінювати моделі складних систем та процесів, розуміння методів зберігання, обробки, вилучення, аналізу та захисту даних; уміння балансувати між протилежними вимогами (наприклад, продуктивність проти безпеки, вартість проти надійності); здатність декомпонувати складні, невизначені проблеми та пропонувати системні рішення; навички ефективної співпраці у мультидисциплінарних та розподілених командах; здатність чітко пояснювати складні технічні концепції як технічним спеціалістам, так і нетехнічним стейкхолдерам (усно, письмово, через документацію); здатність самостійно опановувати нові мови, фреймворки та інструменти в умовах швидкої зміни технологій [213]. В означеному контексті ми повністю підтримуємо позицію С. Петренка щодо того, що «європейські стандарти надають цінні рамки для розробки освітніх і навчальних програм, що відповідають сучасним вимогам галузі та сприяють формуванню висококваліфікованих ІТ-фахівців» [109, с. 215].

З урахуванням наукових підходів до сутності поняття «методологія», «методологічна компетентність», особливостей професійної діяльності фахівців сучасної ІТ-сфери визначаємо **методологічну компетентність майбутніх фахівців з комп'ютерної інженерії** як *компонент професійної компетентності, динамічне інтегративне особистісне утворення, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проєктування для розв'язання складних технічних задач*. Така компетентність виступає регулятивом професійної діяльності, що дозволяє фахівцям не лише адаптуватися до швидкої зміни

інструментарію, а й самостійно конструювати нові алгоритми розв'язання складних науково-технічних задач на основі системного аналізу.

Для визначення структури методологічної компетентності майбутніх фахівців з комп'ютерної інженерії доцільно актуалізувати наукові підходи до компонентного складу методологічної та дослідницької компетентності фахівців різного профілю (Таблиця 1.7).

Таблиця 1.7

**Підходи до визначення компонентів методологічної компетентності
фахівців різного профілю**

Автор	Компоненти
Методологічна компетентність	
М. Опачко (2009) [105, с. 275-276]	(майбутнього вчителя фізики) – когнітивний (знання, що охоплюють виокремленні три рівні методології фізики: знання загальнофілософського характеру: філософські категорії, принципи, філософські ідеї, світоглядні та методологічні переконання вчених; знання загальнонаукового рівня розвитку фізичної науки: загальнонаукові принципи (об'єктивності, системності, логічної структурованості тощо) і методи пізнання (спостереження, експеримент, абстрагування, ідеалізація, порівняння, аналогія, моделювання, гіпотеза, мислений експеримент, математичні методи дослідження); знання конкретно-наукового рівня (фундаментальних наукових понять, фундаментальних взаємодій, фундаментальних експериментів, теорій, констант, методів, законів та закономірностей розвитку фізичної науки); <i>операційний</i> (уміння: дидактичного характеру (визначення ролі і місця методологічного знання в системі навчання, системі уроків; уміння застосовувати методи формування наукового світогляду учнів, системи їх методологічних переконань; уміння застосовувати закономірності формування понять, теорій, фізичних законів); методичного характеру (уміння застосовувати наукові методи на конкретному фізичному матеріалі: ідеалізації, моделі, абстракції тощо; уміння з'ясовувати межі застосовності ідеалізацій, моделей, законів, теорій тощо; уміння застосовувати наукові методи для перевірки і спростування наукових гіпотез: теоретичних (якісної і кількісної оцінки) і практичних (спостереження, експеримент, розв'язування задач); уміння використовувати засоби наочності (карти, схеми, графіки, рисунки, комп'ютерні моделі фізичних явищ, процесів та ін.), зокрема, фізичні прилади для демонстрації фізичних явищ, законів тощо); технологічного характеру (уміння використовувати прийоми, спрямовані на систематизацію, узагальнення, класифікацію, розрізнення, порівняння, інтеграцію знань міжпредметного характеру тощо
І. Козубцов (2014) [53, с. 160]	(аспірантів системи вищої військової освіти) – методологічні знання, уміння, навички з наукової, науково-педагогічної та військово-професійної діяльності, досвід організації діяльності, цінності і ставлення

Н. Кугай (2014) [64, с. 168]	<i>(майбутнього вчителя математики)</i> – наявність системи методологічних знань, умінь і навичок; стійкого інтересу до пізнавальної діяльності; здатності мислити, самостійно розв’язувати світоглядні, дослідницькі, творчі задачі теоретичного чи прикладного характеру в різних сферах життєдіяльності
О. Лаврентьева (2014) [69, с. 251]	<i>(майбутнього вчителя природничих дисциплін)</i> – когнітивна складова – методологічні знання, операційно-технологічна – методологічні уміння, індивідуально-психічні особливості, зокрема методологічні здібності, які підсилюють ефективність методологічного аспекту професійно-педагогічної діяльності
Лі Ліцюань (2020) [75, с. 83]	<i>(майбутніх учителів музичного мистецтва)</i> – мотиваційно-ціннісний, інформологічно-дослідницький, світоглядно-загальнонауковий, конкретно-науково-технологічний, рефлексивно-самовдосконалювальний
Дослідницька компетентність	
Л. Бондаренко (2015) [8, с. 56]	<i>(викладача ЗВО)</i> – мотиваційно-ціннісний (система спонукань та цінностей, що зумовлюють спрямування активності викладача на ефективне здійснення дослідницької діяльності), гностичний (комплекс методологічних, професійних, міждисциплінарних наукових знань і пізнавальних умінь дослідницької діяльності), практично-діяльнісний (сукупність інтелектуальних, практичних, організаторських умінь та здатностей педагога ефективно здійснювати дослідницьку діяльність), особистісно-рефлексивний (творчі, ділові якості дослідника, здатності саморегуляції, самоорганізації власної наукової роботи)
С. Сисоєва, Л. Козак (2016) [132, с. 11]	мотиваційно-ціннісний, когнітивний, процесуально-діяльнісний, інформаційно-комунікаційний, комунікативний, особистісно-творчий, професійно-рефлексивний
М. Вінник (2016) [17, с. 62-63]	<i>(майбутніх інженерів-програмістів)</i> – мотиваційно-ціннісний (наявність інтересу до науково-дослідницької діяльності, прояв провідних мотивів пізнавальної діяльності та системи науково-дослідницьких ціннісних орієнтацій); когнітивний (наявність системи знань інтегративного характеру (фахові, науково-дослідницькі та методологічні знання); діяльнісно-рефлексивний (багатофункціональні пізнавальні вміння, що необхідні для розв’язання професійних завдань, досягнення мети та якісного виконання професійної діяльності; конструктивного ставлення до критики та самокритики, оцінювання та презентації власного досвіду й досягнень)
М. Сорокопуд (2016) [142, с. 88]	<i>(бакалаврів з комп’ютерної інженерії)</i> – мотиваційно-особистісний (мотивація та пізнавальна активність, здатність до подолання когнітивних труднощів; самостійність у процесі пізнання), інтелектуально-творчий (розвиток пізнавальних процесів і навчальних навичок, рівень інтелекту, експериментальне мислення, рефлексивні здібності, загальний рівень креативності, прояв креативності в проблемній ситуації), креативний (знання сутності і технологій основних методів дослідження), діяльнісно-операційний (бачення проблеми, постановка питань, висунення гіпотези, оволодіння навичками проведення експериментів, уміння структурувати матеріал).
О. Біда (2017) [5, с. 11]	мотивації і стимулювання діяльності, організації пізнавальної діяльності, координуючий та комунікативний

Таким чином, аналіз наукових підходів до визначення структури методологічної та дослідницької компетентностей фахівців різних галузей свідчить про відсутність єдиного погляду на їхній компонентний склад, що зумовлено специфікою відповідної професійної діяльності. Відзначимо, що *спільним для більшості дослідників є виокремлення мотиваційного, когнітивного, процесуально-діяльнісного та рефлексивного елементів*. Саме тому при обґрунтуванні структури методологічної компетентності майбутніх фахівців з комп'ютерної інженерії необхідно враховувати як загальнонаукові інваріанти, так і галузевий контекст ІТ-сфери.

Структура будь-якого складного утворення залежить від функцій, які є основою існування будь-якої системи та умовою її виникнення. Відповідно, усвідомлення та характеристики структури передбачає виокремлення функцій, які актуалізують зв'язок структури і мету, структури і завдання.

У філософському контексті функція (від лат. *functio* – виконання, здійснення) потлумачується, як «роль, яку виконують різні структури та процеси у підтримці цілісності та стійкості тих систем, частинами яких вони є» [157, с. 1106]. Визначення функцій передбачає врахування цільових ознак певного виду діяльності, співвідношення між засобами і метою у межах такої діяльності.

У професійній діяльності сучасних фахівців з комп'ютерної інженерії методологічна компетентність виконує такі функції як когнітивна, нормативно-регулятивна, проєктно-конструктивна, рефлексивна, прогностична, преадаптивна та комунікативна.

Когнітивна функція пов'язана із забезпеченням процесу здобуття нових знань про об'єкти комп'ютерної інженерії, здатністю інженера не просто накопичувати технічну інформацію, а опановувати самі методи пізнання, аналізувати логіку розвитку галузі. Це передбачає глибоке розуміння гносеологічних аспектів професії, де фахівець усвідомлює причинно-наслідкові зв'язки між еволюцією архітектур, наприклад, переходом від локальних обчислень до Edge Computing, та здатний критично оцінювати життєздатність нових протоколів ще до їх масового впровадження. Ця функція забезпечує

інтелектуальну гнучкість, дозволяє інженеру не лише освоювати нове обладнання, а й формувати нові стратегії розробки в динамічному технологічному середовищі.

Нормативно-регулятивна функція методологічної компетентності фахівця з комп'ютерної інженерії виступає як внутрішній та зовнішній регулятор професійної діяльності, що визначає межі допустимого, раціонального та етичного в процесі створення складних систем, перетворює інженерну працю на структуровану діяльність, підпорядковану жорстким правилам технічної сумісності, інформаційної безпеки та якості. Методологічна компетентність у цьому контексті дозволяє інженеру не просто механічно дотримуватися інструкцій, а розуміти логіку нормотворення, що дає змогу ефективно діяти в правовому полі інтелектуальної власності та міжнародних стандартів інженерії програмного й апаратного забезпечення. Яскравим прикладом реалізації нормативно-регулятивної функції є впровадження системи менеджменту інформаційної безпеки згідно зі стандартом ISO/IEC 27001 при проектуванні IT-інфраструктур. Фахівець з комп'ютерної інженерії використовує цей стандарт як фундамент для розробки політик доступу, криптографічного захисту даних та фізичної безпеки серверів. Іншим прикладом є використання галузевих стандартів функціональної безпеки, таких як IEC 61508, що регулюють кожен етап життєвого циклу розробки – від аналізу ризиків до верифікації коду. Тут нормативна функція виконує роль етичного та професійного фільтра, який не дозволяє інженеру ігнорувати процедури тестування заради пришвидшення релізу, забезпечуючи високу надійність критично важливих обчислювальних систем.

Проектно-конструктивна функція забезпечує трансформацію теоретичних знань та методологічних принципів у конкретні алгоритми, моделі та технічні рішення. Особливості реалізації цієї функції пов'язані із необхідністю інтелектуального моделювання майбутнього продукту чи системи, у процесі якого фахівець з комп'ютерної інженерії визначає оптимальну послідовність дій, забезпечує декомпозицію відповідних проблем, вибір доцільних інструментальних засобів розробки та синтез відповідних компонентів у цілісну

структуру. Отже, на етапі проектно-конструктивної діяльності здійснюється випереджувальне відображення дійсності, аналіз перебігу функціонування системи з метою мінімізації суперечностей та запобіганню витрату надмірних ресурсів.

Рефлексивна функція пов'язана із необхідністю аналізу професійної діяльності, вчасного виявлення помилок та недоліків в алгоритмах чи технологічних рішеннях. Відповідно, мають бути задіяні механізми самоаналізу та критичної оцінки наявних та перспективних рішень, використовуваних методів, що сприяє інтелектуальній автономії фахівця, формуванню здатності до оцінки меж ефективності використання певних технологій, верифікації професійного досвіду, розумінню значущості новітніх трендів в ІТ-галузі. Підкреслимо актуальність рефлексивної функції для формування критичного мислення ІТ-фахівців, готовності працювати з новими інструментами, виявлення недоліків у конкретних технологічних кейсах.

Прогностична функція методологічної компетентності пов'язана із необхідністю передбачати розвиток технічної системи та наслідки впровадження певної технології. Сутність цієї функції виявляється у здатності до розуміння логіки розвитку галузі, тенденцій розвитку комп'ютерних систем, мереж та методів обробки даних, формуванні навичок стратегічного планування, преадаптації до можливих змін у змісті та технологіях такої галузі як комп'ютерна інженерія.

Надзвичайно важливою функцією методологічної компетентності фахівців з комп'ютерної інженерії є *преадаптивна*, що пов'язано із готовністю до роботи в умовах невизначеності та очікування нових технологічних стандартів. На відміну від простої адаптації, яка є реакцією на вже змінені умови, преадаптація – це стратегічна надлишковість знань, умінь та навичок, що базується на методологічній гнучкості. Так, у контексті підготовки майбутніх учителів Н. Коваленко підкреслює, що «фактором преадаптації виступає випереджальне відображення дійсності, образ «потребового майбутнього» [48, с.166]. Ключові кваліфікації, що зумовлюють готовність до професійної діяльності, «мають широкий радіус

дії, виходять за межі однієї групи професій, професійно і психологічно готують спеціаліста до зміни й освоєння нових спеціальностей і професій, забезпечують готовність до інновацій у професійній діяльності» [48, с. 166].

Преадаптаційна функція методологічної підготовки майбутніх фахівців з комп'ютерної інженерії набуває особливої актуальності в умовах стрімкого розвитку цифрових технологій, постійного оновлення програмних платформ та зміни професійних вимог. Відповідно, необхідно формувати в студентів здатність не лише реагувати на вже наявні зміни, а й бути готовими до появи нових технологічних викликів, які ще не мають остаточно сформованих інструментів чи стандартів застосування. Функційним синонімом преадаптації в інженерії та бізнесі можна вважати поняття «антикрихкості», уведене Н. Талемом (N. Taleb). Також необхідно відзначити, що у межах комп'ютерної інженерії важливим є відмова від жорсткої спеціалізації на користь універсальних принципів побудови обчислювальних систем, що дозволяє фахівцю миттєво інтегрувати інновації, які на момент проєктування системи перебували лише на стадії теоретичних концепцій. Відтак, преадаптивна функція забезпечує не просто виживання фахівця на ринку праці, а його лідерство в умовах неперервного технологічного оновлення, перетворює потенційні кризи на можливості для розвитку.

Комунікативна функція пов'язана із здатністю до ефективного обміну ідеями та концепціями у процесі колективного вирішення складних технічних завдань. У такому контексті йдеться не просто про уміння та навички міжособистісного спілкування, а у володінні універсальною мовою технічної документації, здатності транслювати складні інженерні висновки у зрозумілі для різних стейкхолдерів формати. Відповідно, забезпечується конструктивне спілкування між усіма учасниками виробничого циклу, аргументація доцільності використання ресурсів та технологічних рішень з урахуванням специфіки кожної професійної ситуації. У такому контексті особливої ваги набувають вміння та навички командної роботи, розуміння специфіки ролей в ІТ-команді (менеджери проєкту, ІТ-аналітики, розробники, QA (Quality Assurance)-фахівці, які стежать за

якістю продукту, тестують його та виявляють недоліки, які необхідно усунути, UI (User interface,) та UX(User Experience) дизайнери, які зосереджені на ефективній взаємодії між кінцевими користувачами та додатком, фахівці з аналізу даних, фахівці з кібербезпеки, IT-адміністратори) [20, с.77]. Комунікаційні навички, командна робота та співпраця входять до Топ-10 якостей успішного IT-професіонала (<https://www.ernestgordonrecruitment.com/blog/top-10-qualities-of-a-successful-it-professional/>), до переліку загальних компетентностей, зазначених у Стандарті вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» [143].

Таким чином, поєднання здатності до глибокого пізнання галузевих закономірностей і моделювання складних систем із навичками критичного самоаналізу та стратегічного передбачення дозволяє інженеру не лише ефективно діяти в умовах високої невизначеності, а й створювати гнучку, життєздатну інфраструктуру. Зазначені функції дозволяють розглядати професійну діяльність як цілісний інтелектуальний процес, що постає підґрунтям для успішного вирішення поточних та перспективних проблем в IT-галузі.

З урахуванням наведених функцій структуру методологічної компетентності майбутнього фахівця з комп'ютерної інженерії доцільно представити як складне утворення, що презентується сукупністю взаємопов'язаних та взаємозумовлених компонентів — мотиваційний, когнітивний, праксеологічно-технологічний, рефлексивний та комунікативний.

Важливе місце у структурі методологічної компетентності майбутніх фахівців з комп'ютерної інженерії належить **мотиваційному компоненту** як внутрішньому регулятору та стимулу професійного зростання, що зумовлює наявність інтересу до методологічних проблем предметної галузі, усвідомлення значущості фундаментальних методологічних знань та умінь для успішності вирішення професійних завдань та кар'єрного успіху, перехід від зовнішнього стимулювання до внутрішньої детермінації навчальної та фахової діяльності, що ґрунтується на принципах самоактуалізації та професійної автономії.

В умовах стрімкого розвитку ІТ-галузі мотиваційний компонент методологічної компетентності майбутніх фахівців з комп'ютерної інженерії стає критичним чинником адаптивності та преадаптивності, формуванню здатності до інноваційних технологічних рішень, оновлення методів професійної діяльності.

Когнітивний компонент методологічної компетентності майбутніх фахівців з комп'ютерної інженерії включає систему фундаментальних знань про методологію наукового пізнання, теорію систем, методологію комп'ютерно-інженерної діяльності, принципи та логіку побудови обчислювальних процесів. Це не просто база технічних даних, а «знання про знання», що дозволяє майбутньому інженеру розуміти генезу технологій та орієнтуватися у складних інформаційних потоках. Майбутні фахівці мають володіти знаннями з методології пізнання, зокрема загальнонауковими методами (аналіз, синтез, порівняння, абстрагування, індукція, дедукція тощо) та емпіричними методами, характерними для професійної та наукової діяльності в галузі комп'ютерної інженерії (постановка контрольованих експериментів, вимірювання фізичних і програмних параметрів, порівняльний аналіз альтернативних архітектурних рішень та ін.). Особливо актуальними є розуміння сутності та прикладних аспектів системного аналізу, моделювання, принципи верифікації. Успішна професійна діяльність передбачає володіння методологіями проєктування апаратного забезпечення (на основі моделей, САПР (системи автоматизованого проєктування), розробки вбудованих систем (*Embedded Systems*), життєвого циклу та управління проєктами (каскадна модель, Agile (гнучкі методології), Scrum та ін.). Відповідно, методологічні знання набувають інструментальної цінності, що зумовлюють успішне вирішення складних професійних завдань у розгалужених, гетерогенних (різномірних) та нелінійних комп'ютерних середовищах.

У дисертації Н. Кугай методологічне знання розглядається як «знання про знання і методи пізнання» і як «знання про організацію діяльності» [65, с. 153]. Важливим є уточнення про рівневу структуру методологічного знання (філософський, загальнонауковий, конкретно науковий та технологічні рівні) [65, с. 104]. Така структура актуалізує значення загальнофілософської

підготовки здобувачів вищої освіти та опанування методологією відповідної предметної галузі. Освоєння методології як системи знань, як наголошують О.Гречаний та В.Сабадуха, «завжди здійснюється у вигляді збудження й удосконалення особливої інтелектуальної здібності – здібності рефлексивно, тобто ніби збоку, мислити, оцінювати, регулювати процес виробництва продукту власної, але здійснюваної спільно з іншими людьми трудової діяльності – предмету споживання й суспільного зв'язку» [29, с. 106]. Відповідно, «фахівець, який оволодів методологією, здатен рефлексивно спостерігати й контролювати процеси еволюції структур об'єкта й суб'єкта діяльності та еволюцію їхньої взаємодії» [29, с. 108]. Науковці акцентують увагу на необхідності «самовизначення індивіда й вироблення ним позиції осмислення, *що є що і хто є хто* в соціумі в конкретній ситуації процесу діяльності», «включення людини в дискурсивно-комунікативні практики», «формування сучасної наукової картини світу» [29, с. 118, 120, 122-145]. Наведені позиції повною мірою суголосні позиціям фундаменталізації сучасної освіти, яка закладає міцне знанняве підґрунтя для успішного вивчення профільних предметів через глибоке та системне засвоєння теорії, методології та світоглядних аспектів базових наук. З іншого боку, цей процес забезпечує цілісність, узагальненість та внутрішню єдність усього навчального матеріалу, що будується на органічному поєднанні всіх компонентів освітньої програми та має випереджувальний характер щодо розвитку професійної сфери.

У межах когнітивного компоненту принципове значення мають знання майбутніх фахівців з комп'ютерної інженерії щодо загальнонаукових методів пізнання (аналіз, синтез, класифікація, узагальнення, порівняння тощо) та специфічних методів комп'ютерної інженерії. Загальнонаукові методи розглядаються як наскрізний елемент професійної підготовки в цілому та формування методологічної компетентності зокрема. У свою чергу специфічні методи вводяться за принципом концентричного ускладнення (Рис. 1.2.).

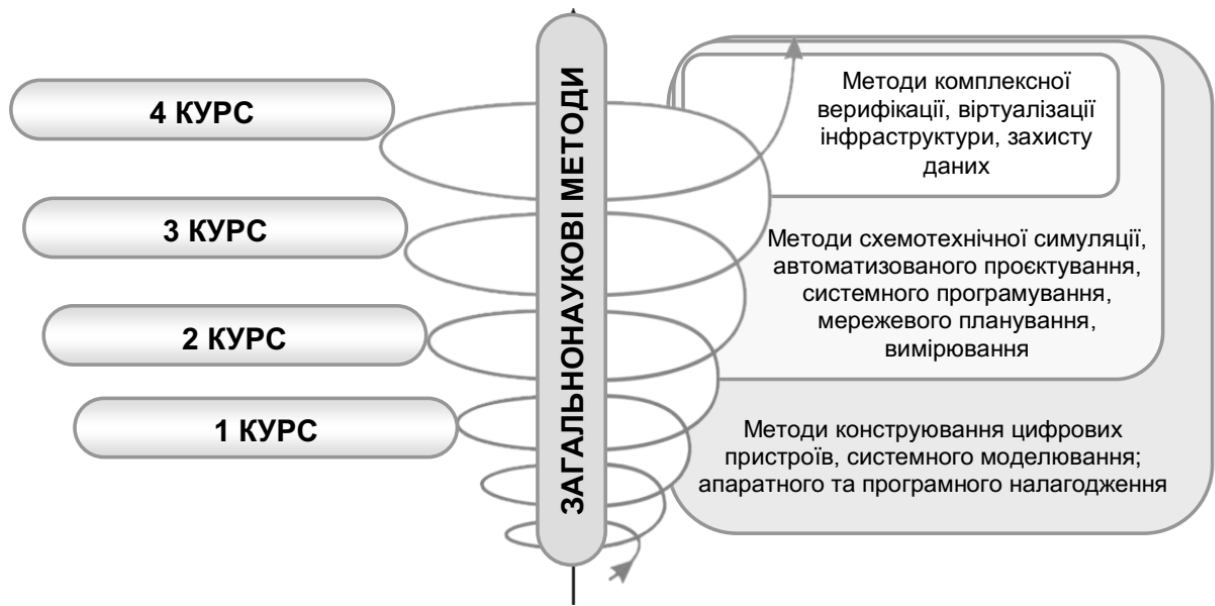


Рис. 1.2. Ускладнення номенклатури методів професійної діяльності у процесі підготовки майбутніх фахівців з комп'ютерної інженерії

Фахівці з комп'ютерної інженерії мають володіти такими *специфічними методами*, як: методи системного та архітектурного моделювання, схемотехнічної симуляції, автоматизованого проєктування та конструювання цифрових пристроїв (CAD/EDA); методи апаратно-програмного копроєктування для розробки вбудованих систем; методи мережевого планування, налаштування протоколів маршрутизації та віртуалізації інфраструктури (зокрема проєктування хмарних і розподілених архітектур); методи системного програмування, паралельних обчислень та інтелектуалізації обчислювальних процесів; методи вимірювання, технічної діагностики й комплексної верифікації (формальної верифікації, апаратного тестування та налагодження, імітаційного й стрес-тестування, оцінки параметрів надійності); а також методи забезпечення кібербезпеки, криптографічного захисту інформації та побудови відмовостійких систем.

Праксеологічно-технологічний компонент корелює з проєктивно-конструктивною, прогностичною, преадаптаційною та комунікативною функціями методологічної компетентності фахівців ІТ-сфери. Цей компонент передбачає володіння загальнонауковими методами пізнання, проблематизації комп'ютерно-інженерної дійсності, методами моделювання, мовами

програмування, засобами автоматизації проєктування та діагностики. Також значущими є навички системного мислення, толерантність до невизначеності, здатність інженера перетворювати абстрактні методологічні схеми в апаратний чи програмний продукт, що забезпечує технологічну спроможність фахівця.

Праксеологія (від грец. *Praxis* – дія та *logos* – вчення) – «програма комплексних наукових досліджень умов, засобів і принципів ефективної та якісної людської діяльності» [157, с. 512], «наука, яка встановлює норми, принципи та закони ефективної діяльності, продуктивної роботи, результативності праці» [119, с. 158]. Як самостійну дисципліну обґрунтував польський філософ Т. Котарбінський [197]. Основні концепти праксеології – оптимізація, доцільність, економічність.

У контексті комп'ютерної інженерії праксеологічний вимір означає не просто володіння технічними навичками, а здатність системно організувати робочий процес, обирати найбільш ефективні алгоритми та приймати прагматичні й виважені архітектурні рішення. Значення праксеологічної спрямованості підготовки ІТ-фахівців, що ґрунтується на фундаментальному та технологічно-прикладному складниках, підкреслює Ю. Тулашвілі [153, с. 377], який зазначає доцільність «активізації квазіпрофесійних дій через реалізацію проєктної діяльності майбутніх ІТ-фахівців» [153, с. 378].

Праксеологічний аспект фокусується на раціональності, доцільності та результативності дій, спрямованих на проєктування, розробку, розгортання та супроводження апаратних і програмних комплексів у межах завдань професійної діяльності фахівців з комп'ютерної інженерії. Відтак, йдеться про готовність інженера діяти усвідомлено й алгоритмічно в умовах швидкої зміни технологічного ландшафту, оптимізувати ресурси та мінімізувати ризики під час розв'язання складних науково-технічних завдань. З іншого боку, технологічний аспект пов'язаний із володінням сучасним інструментарієм, що охоплює середовища розробки, мови програмування високого та низького рівнів, системи автоматизованого проєктування, засоби діагностики, моделювання та тестування комп'ютерних систем. Фахівці з комп'ютерної інженерії мають володіти такими

специфічними методами, як методи системного та архітектурного моделювання, імітаційного моделювання, схемотехнічної симуляції, проєктування та конструювання (CAD/EDA); методи розробки та системного програмування, вимірювання, діагностики та верифікації (формальної верифікації, апаратного тестування та налагодження, стрес-тестування та оцінки надійності); методи забезпечення кібербезпеки та відмовостійкості.

Підкреслимо, що праксеологічно-технологічний компонент виступає сполучною ланкою між фундаментальною методологічною підготовкою та безпосереднім виробничим середовищем, гарантуючи конкурентоспроможність, технологічну гнучкість і готовність інженера до продуктивної професійної діяльності.

Рефлексивний компонент у структурі методологічної компетентності майбутніх фахівців з комп'ютерної інженерії пов'язано із здатністю до самоаналізу, критичного переосмислення методів роботи та готовності до швидкої трансформації професійного досвіду в умовах високої невизначеності. Підкреслимо випереджувальний характер методологічних знань та умінь фахівців ІТ-галузі, що орієнтує не лише на ефективну виконавську діяльність, а на здатність випереджати певні технологічні процеси та оцінювати їх перспективи.

Рефлексія є важливою частиною освітнього процесу, оскільки допомагає усвідомити власні сильні та слабкі сторони, що, в свою чергу, стимулює досягнення професійних цілей. Саме рефлексивні уміння та навички сприятимуть формуванню здатності змінювати свої дії та стратегії в умовах невизначеності та нових викликів. У світі комп'ютерної інженерії технології змінюються стрімко, і способи вирішення задач часто вимагають нестандартного підходу.

Комунікативний компонент методологічної компетентності забезпечує реалізацію відповідних функцій через володіння методами професійної взаємодії та стратегічного планування. Сутнісні характеристики цього компоненту пов'язані із поєднанням глибокого розуміння технічних процесів та здатності

ефективно транслювати, обговорювати й узгоджувати ці знання у професійному середовищі. В ІТ-сфері цей компонент виходить далеко за межі звичайного повсякденного спілкування, оскільки забезпечує спроможність майбутнього фахівця перекладати складні технічні концепції, алгоритми та архітектурні рішення мовою, зрозумілою для замовників, менеджерів та кінцевих користувачів, які не мають профільної освіти. Водночас цей компонент визначає якість внутрішньокмандної взаємодії, де фахівець повинен чітко формулювати технічні завдання, аргументувати вибір певних методологій розробки, критично оцінювати пропозиції колег та вести конструктивний професійний діалог під час проєктування комп'ютерних систем чи програмування.

Важливою ознакою комунікативного складника є його орієнтація на роботу в полікультурному та міжнародному цифровому просторі, що вимагає від інженера не лише досконалого володіння фаховою термінологією та іноземними мовами, а й розуміння специфіки культури розробки, правил ведення технічної документації та стандартів ділового листування. Через призму методологічної компетентності комунікація розглядається як засіб рефлексії та обміну досвідом, де фахівець здатний не просто передавати інформацію, а аналізувати комунікативні бар'єри, оптимізувати інформаційні потоки всередині ІТ-проєкту та застосовувати сучасні інструменти спільної роботи. Таким чином, цей компонент інтегрує когнітивні, мовленнєві та соціально-психологічні навички фахівця, перетворюючи комунікацію на гнучкий методологічний інструмент, який мінімізує ризики помилок у проєктуванні, підвищує ефективність командної роботи та безпосередньо впливає на успішність реалізації високотехнологічних рішень у сфері комп'ютерної інженерії.

Таким чином, структура методологічної компетентності постає як органічна єдність знань, дій, цінностей та здатності до передбачення, що в сукупності визначає конкурентоспроможність фахівця з комп'ютерної інженерії в цифрову епоху.

Формування методологічної компетентності майбутнього інженера з комп'ютерної інженерії є складним процесом, успішність якого значною мірою

залежить від внутрішніх психологічних та інтелектуальних властивостей особистості. Серед рис, що виступають потужними каталізаторами цього процесу, на першому місці стоїть системне мислення, яке дозволяє бачити за окремими компонентами коду чи апаратного забезпечення цілісну структуру взаємозв'язків. Високий рівень інтелектуальної лабільності, тобто здатність швидко перемикатися з одного способу вирішення задачі на інший, сприяє преадаптації та легкому засвоєнню нових методологічних парадигм. Важливу роль відіграє також прогностичність особистості – схильність до передбачення наслідків своїх рішень, що є критично важливим для архітектурного проєктування. Заслуговує на увагу дослідження Д. Філоненко щодо особистісного портрету представника ІТ-сфери, в якому підкреслюється, що важливим рисами таких фахівців є «професійна та особистісна мобільність, автономність, стійкість до невизначеності, відкритість змінам» [156, с. 106].

Водночас необхідно враховувати й певні індивідуальні особливості, які суттєво уповільнюють становлення професіоналізму фахівця. Одним із головних бар'єрів є когнітивний догматизм, що проявляється у схильності до використання шаблонів та небажанні переглядати застарілі знання навіть за наявності доказів їхньої неефективності. Ригідність мислення, як протилежність гнучкості, заважає студенту сприймати альтернативні методологічні підходи, замикаючи його в межах одного знайомого інструментарію. Певним чином до уповільнювального чинника можна віднести й імпульсивність особистості фахівця, яка може його підштовхувати до негайного написання коду без попереднього етапу проєктування та аналізу, що нівелює значення проєктивно-конструктивної діяльності. Крім того, розвитку рефлексивної складника також заважають низький рівень автономності та надмірний конформізм, зайва довіра до авторитетів або некритичне слідування популярним трендам професійної галузі. Отже, особистісний профіль доцільно розглядати як своєрідний фільтр, завдяки якому відбувається трансформація професійних знань та умінь з перспективою на виважене методологічне забезпечення власної діяльності.

У сучасній науковій літературі теоретичний базис дослідження методологічної компетентності характеризується значною поліаспектністю, що зумовлено розмаїттям методологічних підходів до реалізації завдань професійної діяльності. Основними підходами у межах дослідження визначено системний, діяльнісний, особистісно зорієнтований, компетентнісний та праксеологічний.

Відповідно до положень **системного підходу** (В. Андрущенко, С. Гончаренко, В. Кушнір, І. Малафійк, С. Сисоєва та ін.) методологічна компетентність трактується як ієрархічно організована система взаємопов'язаних компонентів, що забезпечують цілісність інженерного мислення. Теоретичним підґрунтям тут виступає загальна теорія систем, яка визначає компетентність не як набір навичок, а як інтегративну властивість особистості, що дозволяє бачити об'єкт проєктування у всій повноті його внутрішніх і зовнішніх зв'язків. Системний підхід акцентує увагу на структурі самої компетентності та її місці у загальному професіоналізмі, розглядаючи її як стрижневий елемент, що координує інші спеціальні компетенції. С. Сисоєва та Т. Кристопчук підкреслюють, що принцип системності «передбачає розгляд об'єкту вивчення як системи: виявлення певної множини її елементів (виділити і врахувати всі їх неможливо, та це й не вимагається), встановлення класифікації і впорядкування зв'язків між цими елементами виділення з множини зв'язків системотвірних, тобто тих, що забезпечують поєднання різних елементів у систему. Системний підхід виходить з того положення, що специфіка складного об'єкту (системи) не вичерпується особливостями елементів, які її складають, а пов'язана передусім з характером взаємодії між елементами. Тому на перший план виходить завдання пізнання характеру і механізму цих зв'язків і відношень, зокрема відношень людини і суспільства, людей всередині певного співтовариства» [131, с. 70].

Комп'ютерна інженерія є високотехнологічною галуззю, що розвивається стрімкими темпами, де знання інструментарію застаріває швидше, ніж завершується цикл навчання студентів. Відповідно до положень системного підходу процес навчання постає як складна відкрита система, що перебуває в постійній взаємодії із зовнішнім середовищем – ринком праці,

технологічними трендами та світовими стандартами ІТ-індустрії. Унаочнення включеності процесу формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії у системи більш високого рівня представлено на Рис. 1.3.

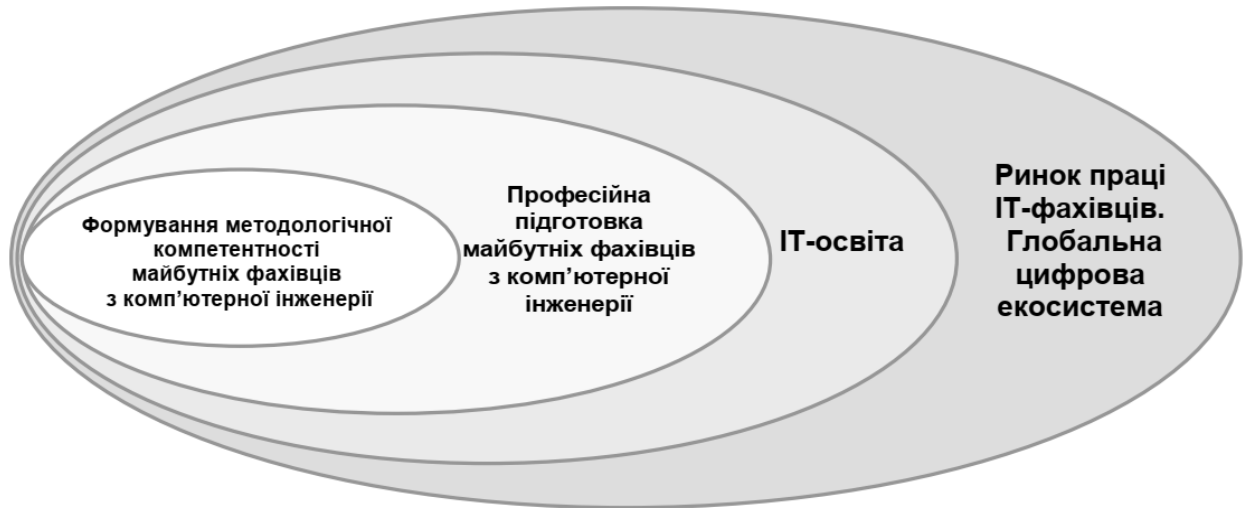


Рис. 1.3. Метасистемний аспект формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії

Необхідність акценту на метасистемному аспекті зумовлена впливом надсистем, зовнішніх чинників на досліджуваний процес. Такими надсистемами виступають – цілісна система професійної підготовки майбутніх фахівців відповідно до державних стандартів, загальні тенденції розвитку ІТ-освіти, вимоги сучасного ринку праці, цифровізація усіх сфер життєдіяльності суспільства. Відповідно процес формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії має враховувати теоретико-методичні позиції щодо: цифрової трансформації, модернізації освіти в цифровому вимірі – В. Биков, О. Спирін, О. Пінчук [3], Н. Морзе, О. Буйницька [94]; інформатизації освіти – В. Биков, О. Буров, М. Жалдак, С. Литвинова, В. Олійник та ін. [149]; формування цифрової культури та цифрової компетентності майбутніх фахівців, використання цифрових освітніх середовищ – Л. Гаврілова, Я. Топольник [19; 151], Т. Собченко, О. Кін, С. Кириленко [141] та ін.

Системний підхід у реалізації завдань дослідження дає змогу проєктувати зміст освіти, педагогічні технології та методи контролю як взаємопов'язані підсистеми, спрямовані на досягнення єдиного синергетичного ефекту. Кожен елемент навчального плану – від фундаментальної математичної підготовки до практичного проєктування апаратних засобів – починає працювати на спільну мету, де якість кінцевого результату перевищує просту суму властивостей окремих частин. Крім того, системний підхід забезпечує процесу формування компетентності необхідну гнучкість та адаптивність, оскільки він орієнтований на виявлення зворотних зв'язків. Регулярний моніторинг результатів навчання та корекція педагогічних впливів на основі системного моделювання дозволяють оперативно реагувати на появу нових технологічних викликів.

У підсумку, саме системна методологія дає можливість не просто зафіксувати поточний стан підготовки студентів, а спроектувати цілісну та життєздатну дидактичну модель, яка гарантує формування конкурентоспроможного фахівця, здатного до системного мислення у своїй майбутній інженерній практиці.

Діяльнісний підхід зумовлює розгляд методологічної компетентності майбутніх фахівців з комп'ютерної інженерії в контексті практико-зорієнтованої природи цієї професії. Концептуально-теоретичними засадами цього підходу є положення про єдність свідомості та діяльності, де компетентність постає як здатність фахівця перетворювати теоретичне знання на алгоритм практичних дій. За словниковими визначеннями діяльність – це «форма активності, що характеризує здатність людини чи пов'язаних з нею систем бути причиною змін у бутті» [157, с. 163], «спосіб буття людини в світі, здатність її вносити в дійсність зміни» [25, с. 98], «динамічна система взаємодій суб'єкта зі світом, у процесі яких відбувається виникнення і втілення в об'єкті психічного образу і реалізація опосередкованих ним відносин суб'єкта предметній дійсності» [164, с. 101]. Діяльнісний підхід розглядається як «конкретно-наукова методологія для психології вікової, педагогічної, інженерної, медичної та ін.» [164, с. 336], а

також як «система переходів від освітньої діяльності академічного типу (лекція, аудиторне заняття, чи семінар за допомогою Zoom, Google Meet) через форми квазіпрофесійної діяльності до освітньо-професійної (тренінг, курсова робота, проєктні завдання), а надалі – до реальної професійної діяльності» [68, с. 142]. Надзвичайно актуальною є позиція І. Беґа стосовно того, що «діяльнісний підхід дає відповідь на запитання: як побудувати процес засвоєння знань так, щоб він одночасно був процесом формування здібності мислити» [2, с. 29].

У монографії «Діяльнісні засади підготовки майбутніх компетентних фахівців в умовах сучасних викликів» (Житомир, 2024) зазначено, що діяльнісний підхід зумовлює логіку підготовку фахівців – «від формування інтеграційних якостей до їх конкретизації, освоєння і відображення в індивідуально-творчих нормах, а далі відбувається освоєння окремих операцій, функцій, видів діяльності до їх узагальнення» [30, с. 22]. На значущості діяльнісного підходу у формуванні професійної компетентності майбутніх фахівців ІТ-сфери наголошують такі дослідники, як К. Стрюк [145, с. 89], М. Вінник [17, с. 94-95] та ін.

Важливо підкреслити, що особливості діяльнісного підходу у формуванні методологічної компетентності майбутніх фахівців з комп'ютерної інженерії пов'язані із фокусом на операційному аспекті – вмінні інженера використовувати методи аналізу, синтезу та моделювання для отримання конкретного технічного результату. У сфері інформаційних технологій та апаратного проєктування теоретичні знання самі по собі не гарантують успіху, оскільки професійна спроможність інженера виявляється виключно в дії, під час розробки архітектури систем, написання коду, тестування чи розв'язання складних технічних проблем. Тому методологічна компетентність, яка передбачає здатність організовувати, оптимізувати та трансформувати саму дослідницьку та виробничу діяльність, не може бути сформована через пасивне сприйняття інформації, а вимагає активного включення студента у процеси, що моделюють його майбутню роботу. Діяльнісний підхід дозволяє розглядати процес навчання як динамічну струк-

туру, де засвоєння знань відбувається не у вигляді ізольованих фактів, а через інтерналізацію способів діяльності, методів проектування тощо. У контексті комп'ютерної інженерії це означає, що майбутній фахівець опановує методологію дослідження складних систем лише тоді, коли сам проходить крізь усі етапи інженерного пошуку – від аналізу суперечливих вимог замовника до верифікації готового продукту. Такий підхід зміщує фокус з того, що студент знає, на те, як він діє в умовах невизначеності, як формулює гіпотези, вибирає інструментарій розробки та оцінює ефективність створених ним системних рішень.

Використання діяльнісної парадигми забезпечує трансформацію навчальної діяльності студента у професійну діяльність інженера через послідовне ускладнення вирішуваних завдань. Відповідно, у процесі навчання кожен теоретичний блок має підкріплюватися квазіпрофесійною діяльністю (лабораторними експериментами, наскрізним курсовим проектуванням чи командною розробкою проєктів (Scrum, Agile). У процесі такої активності майбутні фахівці стикаються з реальними технологічними обмеженнями, вчаться рефлексувати над власними помилками, оптимізувати алгоритми та адаптуватися до нових інструментів, що безпосередньо стимулює розвиток їхнього методологічного мислення. Також відзначимо, що діяльнісний підхід сприяє актуалізації мотиваційного складника компетентності, оскільки будь-яка діяльність спонукається певними мотивами та має чітку цілепокладальну основу.

Особистісно зорієнтований підхід (І. Бех, І. Зязюн, О. Пехота та ін.) у професійній підготовці майбутніх фахівців з комп'ютерної інженерії пов'язано із трансформацією орієнтації із пасивного засвоєння готових знань на перетворення майбутнього фахівця з комп'ютерної інженерії на активного суб'єкта власного професійно-інтелектуального розвитку. Такий підхід забезпечує врахування індивідуально-психологічних особливостей, пізнавальних інтересів та особистісного потенціалу здобувача освіти, що дозволяє формувати методологічну компетентність через побудову індивідуальної освітньої траєкторії, стимулювання рефлексивного мислення та розвитку здатності до самостійного прийняття рішень у складних інженерних ситуаціях.

Відповідно, методологічна компетентність із сукупності абстрактних алгоритмів перетворюється на особистісно значущий інструментарій, завдяки якому майбутній інженер набуває готовності свідомо обирати методологічний інструментарій, критично оцінювати власну дослідницько-проектувальну діяльність та адаптуватися до стрімких технологічних змін протягом усього життя.

Одним із важливих методологічних підходів у контексті дослідження є **компетентнісний підхід**, концептуальні засади якого у вищій освіті представлено у підрозділі 1.1. Такий підхід орієнтовано на формування здатності здобувачів вищої освіти ефективно діяти, інтегрувати набутий досвід та успішно вирішувати нестандартні завдання у динамічному професійному середовищі.

У контексті підготовки майбутніх фахівців з комп'ютерної інженерії компетентнісний підхід дозволяє розглядати методологічну компетентність як складну індивідуально-особистісну якість, що поєднує в собі не лише когнітивний складник, а й практичний досвід, ціннісні орієнтації, мотивацію та здатність до постійного самонавчання. Використання цього підходу також забезпечує подолання розриву між академічною теоретичною підготовкою у ЗВО та реальними вимогами ІТ-галузі.

Компетентнісний підхід вимагає проектування освітнього процесу «від результату», тобто на основі аналізу професійних стандартів, профілів компетентностей сучасних інженерів та очікувань роботодавців, які потребують не просто носіїв інформації, а фахівців із розвиненим критичним та методологічним мисленням.

Праксеологічний підхід як методологічне підґрунтя дослідження пов'язаний із орієнтацією всього освітнього процесу на досягнення максимальної ефективності, доцільності та раціональності квазіпрофесійної та майбутньої професійної діяльності студентів. Праксеологія як загальна теорія ефективної організації діяльності розглядає будь-яку дію крізь призму її прагматичної цінності, оптимізації витрат ресурсів та отримання гарантованого корисного

результату [197; 92; 119]. Праксеологічний підхід «трактується як організація раціональної, ефективної людської діяльності, як спосіб аналізу досягнення поставленої мети і отримання продукту з мінімальними ресурсними витратами» [92, с. 47].

Досліджуючи праксеологізацію вищої освіти І. Мельничук наголошує, що «вища освіта в контексті праксеологічного підходу є інтегрованим освітнім середовищем підготовки фахівців, що розвивається з урахуванням необхідності найраціональнішого використання ресурсів та індивідуалізованої спрямованості освітньої діяльності.... Вивчення освітнього процесу у вищій школі та його оптимізація з урахуванням праксеологічного знання передбачають таку взаємодію суб'єктів освітньої практики, яка змінює механізми ресурсного забезпечення цього процесу, мінімізує ризикогенні чинники і передбачає нову якість одержуваного освітнього результату (продукту)» [92, с. 47]. Заслужують на увагу такі механізми праксеологізації: поєднання занять теоретичного (знаннєвого) характеру з практикою (практичними заняттями); сингуляризація, яка полягає в тому, що особистість освоює практичні прийоми, що вимагаються не лише для тієї діяльності (праці), яка освоюється, а для більш широкого діапазону життєвих ситуацій; мотивація, що пов'язана із тим, що пропонується сукупність практичних заходів, які прямо не стосуються освоюваної діяльності (праці), але які стимулюють бажання для цього виду робіт» [92, с. 48].

У сфері комп'ютерної інженерії, де розробка складних апаратних і програмних систем завжди лімітована жорсткими рамками часу, обчислювальних потужностей, фінансів та вимог до надійності, праксеологічний вимір стає невід'ємною частиною мислення професіонала. У межах цього підходу зміщується акцент з пасивного вивчення теорії на виокремлення критеріїв успішності та раціональності дій студента під час вирішення інженерних завдань. Це дозволяє досліджувати процес навчання як простір, де майбутній фахівець опановує мистецтво мінімізації помилок, вчиться раціонально розподіляти навантаження на систему, оцінювати архітектурну складність рішень та автоматизувати рутинні процеси розробки.

Сутність праксеологізації підготовки комп'ютерних інженерів виявляється у формуванні в них особливої культури дій, яка базується на праксеологічних принципах – від чіткого цілепокладання та ретельного планування до фінального аналізу ефективності створеного продукту (рефлексії). Відповідно, освітній процес доцільно моделювати як систему послідовних праксеологічних ситуацій, де студент поставлений в умови необхідності оптимізації своїх дій, наприклад, під час вибору між різними архітектурними патернами чи технологіями. У результаті майбутній фахівець вчиться бачити за кожним технічним рішенням його практичну та економічну доцільність, що трансформує теоретичні методологічні знання у дієвий інструмент створення високоякісних ІТ-продуктів. Практичний підхід забезпечує тісний взаємозв'язок між методологічною компетентністю інженера та його індивідуальною продуктивністю в командній роботі, усвідомлення цінності раціональної комунікації, ефективного розподілу ролей та використання сучасних методологій управління проєктами, таких як Agile чи DevOps.

Узагальнення теоретико-методологічних засад дослідження дозволяє зробити висновок, що методологічна компетентність майбутніх фахівців з комп'ютерної інженерії є складним, інтегральним, динамічним особистісно-професійним утворенням, яке визначає готовність і здатність випускника ефективно здійснювати науково-дослідну та проєктно-конструкторську діяльність у високотехнологічній галузі. Структурна організація зазначеної компетентності репрезентована взаємопов'язаними компонентами, кожен з яких виконує свою специфічну функцію в професійній діяльності інженера. Мотиваційний компонент виступає внутрішнім рушієм, що формує прагнення до винахідництва, усвідомлення етики інженерної діяльності та потребу в неперервному професійному розвитку. Когнітивний компонент охоплює систему фундаментальних і спеціальних знань про методологічні принципи, патерни проєктування та системний аналіз, які слугують теоретичним підґрунтям для прийняття рішень. Практично-технологічний компонент виявляється в безпосередньому володінні інструментарієм розробки, здатності моделювати

складні процеси, оптимізувати архітектурні рішення та ефективно працювати в команді за сучасними праксеологічними методологіями. Нарешті, рефлексивний компонент забезпечує здатність до критичної оцінки власних дій, аналізу помилок (рефакторингу мислення) та прогнозування життєвого циклу створюваних комп'ютерних систем.

Синергетичний ефект взаємодії всіх окреслених компонентів забезпечує перехід майбутнього фахівця з комп'ютерної інженерії на суб'єктний рівень професіоналізму, де сформована методологічна компетентність сприяє конкурентоспроможності, здатності до інноваційної творчості впродовж усього життя.

1.3. Технології проблемно зорієнтованого навчання у вищій школі: сутність, моделі, форми та методи

У сучасному, динамічному та швидкозмінному світі, що характеризується глобалізацією, технологічним прогресом та постійно зростаючою складністю викликів, перед системою вищої освіти стоїть завдання не просто передавати знання, а й готувати фахівців, здатних до гнучкості, адаптації та неперервного навчання. Європейською рамкою особистісної, соціальної та навчальної ключової компетентності (LifeComp) (2020) серед ключових компетентностей визначено критичне мислення, здатність до розробки креативних ідей, синтезу та поєднання концепцій та інформації з різних джерел з метою вирішення проблем. Стандартами вищої освіти в Україні серед загальних компетентностей позначається «вміння виявляти, ставити та вирішувати проблеми». Така загальна компетентність, наприклад, наявна в Стандарті вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти [143].

Традиційні методи навчання, що фокусуються на пасивній передачі інформації, часто виявляються недостатніми для формування компетенцій, необхідних для успішної професійної та соціальної реалізації у XXI столітті.

Саме тому дослідження та імплементація проблемно зорієнтованого навчання у вищій школі набуває надзвичайної актуальності.

Концептуальні засади проблемно зорієнтованого навчання репрезентовано в наукових працях таких зарубіжних дослідників, як Г. Барроуз (H. Barrows), Т. Баррет (T. Barrett), Р. Джонс (R. Jones), Х. Маурер (H. Maurer), Дж. Моуст (J. Moust), Р. Тамблін (R. Tamblyn), А. Теланг (A. Telang), С. Хмело-Сільвер (C. Hmelo-Silver) та ін. Сутнісні ознаки моделі проблемно зорієнтованого навчання представлено в студіях Г. Барроуза [170; 171; 172], С. Хмело-Сільвер [192], ключові принципи навчання – А. Теланг [215] та ін.

Г. Барроуз (H. Barrows), Р. Тамблін (R. Tamblyn) розглядають проблемно зорієнтоване навчання в медичній освіті, наголошують на двох його фундаментальних постулатах: навчання через вирішення проблем набагато ефективніше для створення бази знань, які можна буде використовувати в майбутньому, і що лікарські навички, найважливіші для пацієнтів, – це навички вирішення проблем, а не навички пам'яті [170]. У контексті впровадження досліджуваного типу навчання у вищу освіту зарубіжні автори акцентують увагу на питаннях поєднання проблемно зорієнтованих практик з освітньою теорією, навчання з великими групами, витрат часу та задоволеності студентів [178].

Т. Баррет розглядає проблемно зорієнтоване навчання як гнучкий підхід, який може бути адаптований до різних дисциплін та контекстів. Особливу увагу приділено студентоцентрованому підходу, студентському дискурсу в процесі навчання, міждисциплінарності та інтеграції знань [169].

Українські науковці досліджують теоретико-методичні засади проблемно зорієнтованого навчання в процесі підготовки майбутніх фахівців медичної галузі (О. Беляєва, Л. Горяча, І. Коломієць, Т. Прокопенко, Л. Хламанова, Л. Хмель, Ю. Чайковський, Л. Хмель та ін.), учителів іноземних мов (С. Боднар, Ю. Будас, Н. Дмітренко, Т. Глушко та ін.). Відповідно до завдань підготовки майбутніх фахівців програмної інженерії О. Баранюк відзначає роль викладача, який керує процесом навчання на основі власного досвіду та запропонованого набору проблем [1].

У вітчизняному педагогічному дискурсі традиційно використовується поняття «проблемне навчання», яке потлумачується як «один з видів розвиваючого навчання, істотною відмінністю якого є зближення психології мислення людини з психологією навчання. Проблемне навчання найповніше відповідає завданням розвитку творчого мислення учнів (студентів)» [25, с. 271]. Концептуальні засади проблемного навчання представлено в працях таких відомих дослідників, як І. Лернер, О. Матюшкін, М. Махмутов, Л. Момот, В. Оконь, В. Онищук, А. Фурман та ін.

Сутність проблемного навчання полягає в пошуковій діяльності учнів, яка починається з постановки питань, розв'язанні проблем і проблемних завдань, закладених у навчальних програмах і підручниках, у проблемному викладі й поясненні знань учителем, у різноманітній самостійній роботі учнів» [25, с. 271]. Показовим є зауваження польського науковця В. Оконя, який ще в 90-х рр. XX ст. наголошував, що «проблемне навчання «ґрунтується не на передаванні студентам готової інформації, а на набутті знань та розвитку професійних умінь шляхом вирішення теоретичних та практичних проблем» [202].

В. Павленко підкреслює, що «проблемне навчання зосереджується на практичному навчанню студентів, а не на запам'ятовуванні фактів. Воно спонукає студентів до використання навичок мислення високого рівня, які вимагають від них аналізу висловлювання, створення продукту, захисту думки або оцінки судження» [108, с. 100]. На основі аналізу наукових праць І. Лернера та О. Матюшкіна дослідники С. Боднар та Т. Глушко зауважують, що «проблемний виклад означає, що викладач, створивши проблемну ситуацію, не подає фінальний результат вирішення проблеми, а демонструє процес вирішення всіх поставлених завдань» [7, с. 101].

Підкреслимо, що проблемне навчання, обґрунтоване вченими в 70 – 80-х рр. XX ст., переважно розглядалося як метод доцільний та дієвий швидше для середньої, а не вищої школи [79, с. 93]. Більшість посібників / підручників з педагогіки вищої школи, як зауважують автори, визначають суть

проблемного навчання в постановці (педагогом) і розв'язанні (студентом) проблемного питання, завдання і ситуації [79, с. 93].

У монографії О. Кисельової, Л. Коломієць та А. Шевцова «Якість вищої освіти: організація навчання та вимірювання знань» (2017) *проблемне навчання* розглядається як «особлива структура творчої навчальної діяльності студентів щодо засвоєння знань і способів діяльності з наявністю аналізу проблемних ситуацій, формулювання проблем та їх вирішення за допомогою висування припущень, обґрунтування і доказів гіпотез» [44, с. 88], *проблемне викладання* – як «діяльність викладача, яка забезпечує умови проблемного навчання студентів шляхом зумисного створення системи послідовних проблемних ситуацій» [44, с. 87-88]. Автори відзначають такі структурні елементи проблемного навчання: «актуалізація вивченого матеріалу; створення проблемної ситуації; постановка навчальної проблеми; побудова проблемної задачі; розумовий пошук і вирішення проблеми (формулювання гіпотези чи декількох гіпотез, заснованих на припущеннях і уже відомих фактах; доказ гіпотез, аналіз можливих помилок, передбачення природних наслідків з кожної гіпотези, узагальнення); перевірка вирішення проблеми і повторення» [44, с. 87]. Важливим, на нашу думку, є висновок дослідників про те, що «головною лінією проблемного навчання в виші повинна бути установка на розкутість мислення студентів, відхід від зайвої стандартизації в підходах до вирішення різних задач. Природно, ця розкутість жодною мірою не повинна зводитись до рівня заперечення заради заперечення, демагогічного дискутування. Вона має слугувати, насамперед, пробудженню невичерпного творчого потенціалу особистості, в основі діяльності якої лежить творчість» [44, с. 89].

Отже, у вітчизняній психолого-педагогічній літературі усталеними є позиції щодо проблемного навчання як виду навчання, спрямованого на активізацію пізнавальних процесів, формування навичок самостійної дослідницької діяльності учнів / студентів.

У зарубіжному науковому дискурсі з 60-х рр. XX ст. набуває поширення поняття «**проблемно зорієнтоване навчання**» (*Problem-Based Learning*) як:

- навчальний підхід, який допомагає студентам розвивати (удосконалювати) гнучкі знання («*flexible knowledge*»), навички ефективного розв'язання проблем («*effective problem-solving skills*»), навички самонавчання («*self-directed learning*») і навчання впродовж життя («*lifelong learning skills*»), навички ефективної співпраці («*effective collaboration skills*»), а також внутрішню мотивацію («*intrinsic motivation*») [192];
- «навчальний метод, спрямований на розвиток у студентів здатності самостійно вирішувати проблеми, самостійно навчатися протягом усього життя та працювати в команді» [182, с. 654];
- педагогічний підхід, «який дає студентам змогу навчатися, активно вирішуючи в процесі навчання значущі проблеми, застосовуючи когнітивні моделі навчання, а також формувати здатність до самостійного навчання шляхом практики і рефлексії» [223].

Отже, у наукових працях та енциклопедичній літературі проблемно зорієнтоване навчання розглядається як навчальний підхід, навчальний метод, педагогічний підхід, що свідчить про варіативність вихідних позицій дослідників, залежність від контексту реалізації.

Термін «проблемно зорієнтоване навчання» (PBL) введено в науковий обіг у 60-х рр. XX ст. викладачами Університету Мак-Мастера в Канаді, які займалися підготовкою майбутніх фахівців у галузі медицини (Д. Вудс, Г. Барроуз). Г. Барроуз визначив 4 основні цілі проблемно зорієнтованого навчання: структурування знань і клінічного контексту; формування клінічного мислення; формування навичок самостійного навчання; внутрішня мотивація. Відтак, робота студентів у малих групах дає змогу через постійний контакт із клінічними кейсами навчитись оцінювати проблему пацієнта, знайти причину її виникнення і приймати відповідні

клінічні рішення [172]. Наступними медичними школами, які адаптували розроблену модель і впровадили її у своїй практиці, були університет Лімбурга в Маастрихті (Нідерланди), університет Ньюкасла (Австралія) і університет у Нью-Мексико (США) [54, с. 90].

Теоретичними засадами для обґрунтування сутності проблемно зорієнтованого навчання є ідеї експериментально зорієнтованого навчання Дж. Дьюї, теорії конструктивізму Ж. Піаже, соціального конструктивізму Л. Виготського, теорія практичного навчання Д. Колба, концепція рефлектуючого практика Д. Шона.

Модель проблемно зорієнтованого навчання (PBL), запропонована Г. Барроузом, містила такі положення: 1. Навчання за допомогою PBL завжди зосереджено навколо студентів. 2. Учителі використовують малі групи для навчання студентів. 3. Учителів називають фасилітаторами або гідями. 4. Тематичні проблеми стають первинним фокусом, що стимулює навчання. 5. Студенти освоюють навички вирішення проблем за допомогою клінічних проблем. 6. Навчання є самостійним, і студенти вивчають нову інформацію [172].

Відповідно до позицій Г. Барроуза проблемно зорієнтованому навчанню притаманні такі риси:

- Центральне місце курсу чи програми займає складна, слабо структурована, оригінальна проблема.
- Оригінальне завдання прив'язане до реальної життєвої ситуації і потребує від студентів застосування широкого спектра знань і вмінь.
- Процес навчання орієнтований на студентів й очікується, що вони візьмуть на себе ініціативу і відповідальність за навчальне пізнання.
- Індивідуальний підхід до студентів у процесі вирішення проблеми.
- Мультидисциплінарний підхід до вирішення проблем.
- Викладач виконує роль фасилітатора, наставника (тьютора), а не є джерелом знань.

- Студенти працюють у малих групах для того, щоб визначити потреби пізнання і знайти рішення проблем.
- Оцінка досвіду навчання включає оцінку тьютора, самооцінку й оцінку одногрупників.
- Оцінка використовується як інструмент рефлексії та аналізу освітнього процесу [172, с. 10].

С. Хмело-Сільвер (С. Hmelo-Silver) доповнює виклад основних характеристик проблемно зорієнтованого навчання такими позиціями:

- комплексне вирішення завдань, яке ставить за мету постановку і спільне дослідження складних проблем, аналіз та узагальнення вивченого і зібраного самостійно матеріалу з метою віднаходження оптимального шляху і виявлення можливих варіантів вирішення завдань;
- достовірність і реалістичність навчання, спрямованого на реалізацію таких проєктів, які становлять інтерес для сучасного суспільства, науки і освіти;
- налаштування на співробітництво, що зумовлено необхідністю спільного виконання завдань, рішенням складних завдань, установленням партнерських взаємин з викладачем [192, с. 244].

У межах проблемно зорієнтованого навчання дослідники виокремлюють «чотири ключові принципи навчання для PBL – це конструктивне, спільне, контекстне та самостійне навчання» [215]. Саме конструктивістські ідеї навчання зумовлюють акцент на, по-перше, активній участі студентів, які визначають наявні знання, зв'язують їх із новою інформацією; по-друге, співпраці в навчанні [215].

У досліджуваному виді навчання саме проблема є центральною категорією, передумовою та фокусом освітньої діяльності. *Проблема* (від гр. *problema* – перешкода, утруднення, завдання) – питання, що об'єктивно виникає в ході пізнання, або комплекс питань, вирішення яких становить істотний практичний або теоретичний інтерес. Етимологію терміна «проблема» прийнято вести від грецького дієслова *πρόβλημα* («балейн») –

кидати, тобто проблема – це «об’єкт, що кинутий уперед». У повсякденному мовленні поширено кліше «немає проблем», «без проблем», «це не моя проблема», «це твої проблеми» і т.п. За такими варіантами слово проблема стає синонімом перешкод та утруднень у житті [10, с. 21–22].

У контексті зарубіжних досліджень для роботи із проблемами пропонуються такі кроки [201, с. 22):

Крок 1. Уточнення незрозумілих термінів та понять у тексті задачі

Крок 2. Визначення задачі: Що саме потрібно пояснити?

Крок 3. Аналіз задачі: Сформулюйте якомога більше ідей.

Крок 4. Аналіз задачі: Систематично впорядкуйте ідеї та проаналізуйте їх детально.

Крок 5. Формулювання навчальних цілей.

Крок 6. Пошук інформації в навчальних ресурсах.

Крок 7. Синтез та застосування нової інформації. У роботі з проблемами різного типу однією з важливих умов є організація невеликої групи студентів, «оскільки взаємний обмін знаннями та ідеями є важливим, навчальна група може складатися лише з обмеженої кількості учасників. Коли групи складаються з понад десяти осіб, можливості для кожного члена групи відігравати активну роль швидко *обмежуються*» [201, с. 52].

Зарубіжні дослідники пропонують до впровадження у вищу освіту різні моделі / стратегії проблемно зорієнтованого навчання:

- **«Pebble in the Pond»** (Г. Барроуз). Сутність метафори «камінь у ставок» полягає в такому: *камінь (Pebble)* – це проблемний випадок, сценарій або питання, яке є відправною точкою для навчання. Він є невеликим, але викликає широкі наслідки; *ставок (Pond)* – це весь обсяг знань, концепцій та навичок, які студенти мають опанувати у певній галузі або курсі; *хвилі (Ripples)* – коли камінь кидають у ставок, він створює хвилі, що розходяться. Такі хвилі символізують процес навчання та дослідження, який ініціюється проблемою. Студенти, досліджуючи проблему, виявляють необхідність у

нових знаннях, шукають інформацію, аналізують її та застосовують для вирішення поставленого завдання.

- **Challenge-Based Learning (CBL)** – навчання через виклик. CBL спрямований на пошук рішень для глобальних, амбітних викликів, що не мають однозначних або готових відповідей. Зазвичай процес CBL розгортається у три взаємопов’язані фази: 1. *Залучення*. Навчання починається з широкої, абстрактної концепції або глобальної теми (наприклад, стале енергоспоживання, зміна клімату, нерівність, здорове старіння). Далі визначення виклику (*Challenge*), коли студенти, часто за підтримки викладача та зацікавлених сторін, перетворюють цю «велику ідею» на конкретний, дієвий і значущий виклик (наприклад, «Як ми можемо зменшити споживання енергії в нашому університеті?»).
- 2. *Дослідження*. У межах цієї фази формулюється низка запитань, які допомагають дослідити проблему з різних боків, проводяться дослідження на основі різних джерел (академічні статті, експертні інтерв’ю, статистичні дані тощо).
- 3. *Завершальна фаза – дія* (розробка рішення у вигляді певного продукту, рекомендацій і т.п.; упровадження та тестування на основі реальної взаємодії із стейкхолдерами та спільнотою; презентація рішень та рефлексія над процесом навчання, над здобутими знаннями та навичками).

- **4C/ID (Four-Component Instructional Design)** – це фреймворк для розробки освітніх програм у контексті проблемно зорієнтованого навчання. Основні компоненти цієї моделі – *навчальні завдання* (автентичні, цілісні та комплексні завдання / проблеми, які організовані від простих до складних); *підтримувальна інформація* (пошук нових знань та концепцій як теоретична база для розуміння та вирішення проблеми; контекстуалізація знання); *процедурна інформація* як інструктивні приписи для виконання рутинних частин завдання [221].

Існує інший підхід до визначення моделей проблемно зорієнтованого навчання (PBL):

1. *Чисте PBL* як оригінальна форма PBL. У процесі навчання немає лекцій чи подібних форм поширення знань. Студенти беруть на себе

найвищий ступінь відповідальності за управління своїм процесом вирішення проблем та навчання. Завдання є дуже складними, слабо структурованими та максимально автентичними.

2. *Гібридне PBL* як комбінація чистого PBL з обмеженою кількістю лекцій як додаткового навчання. Однак студенти отримують обмежену кількість лекцій або міні-лекцій, щоб забезпечити достатнє охоплення та точність засвоєння знань.

3. *Закріплене навчання* передбачає використання відео-сценаріїв для закріплення вмінь та навичок студентів у реальних життєвих ситуаціях, вимагає вирішення завдань на основі попередніх знань. Нове знання в межах предметної галузі надається вчителем за потреби.

4. *Проектне навчання*. Студентам доручається виконати проєкт, у якому вони повинні розробити рішення реальної життєвої проблеми на основі вивченого матеріалу. Проте процес вирішення проблем функціонує головним чином для застосування знань, а не для їх отримання.

5. *Навчання на основі кейсів* дозволяє встановити зв'язки між теоріями та застосуваннями. Вивчаючи та аналізуючи реальні проблеми / кейси, студенти усвідомлюють, як абстрактні поняття використовуються або проявляються в реальних ситуаціях.

6. *Навчання на основі лекцій із завданнями на вирішення проблем*. Ця модель на основі кейсів має найнижчий ступінь самостійності та структурованості використаних завдань, спрямована на забезпечення зв'язку з теоретичними концепціями [182, с. 655–656].

М. Севін-Бейден (M. Savin-Baden) пропонує свій підхід до виокремлення виділяє моделей проблемно зорієнтованого навчання [209]. Дослідниця відзначає такі моделі:

PBL для епістемологічної компетентності орієнтована на засвоєння фундаментальних теоретичних знань та розуміння способів їх застосування. У межах цієї моделі навчальна проблема розглядається перш за все як інструмент для освоєння предметного змісту конкретної дисципліни. Головна мета полягає у

тому, щоб навчити здобувачів освіти аналізувати неструктуровану інформацію, застосовувати базові концепції для знаходження обґрунтованого рішення. Викладач виступає експертом, який спрямовує аналітичний процес студентів та перевіряє точність засвоєння теорії, оскільки саме знання дисципліни вважається головним результатом навчання.

PBL для професійної діяльності (дії) зміщує фокус із теоретичного знання на формування практичної компетентності («know-how») та готовності діяти у реальних робочих ситуаціях. Проблемні сценарії моделюють конкретні професійні кейси з відповідної професійної галузі, де від студентів вимагається не просто демонстрація знань, а прийняття практичних рішень і виконання конкретних процедур. Основний акцент робиться на розвитку алгоритмів професійного мислення, навичок розв'язання практичних завдань та оцінюванні ефективності дій у середовищі, що імітує професійну практику.

PBL для міждисциплінарного розуміння спрямована на подолання штучних меж між окремими навчальними предметами та поєднання теорії з практикою. У цій моделі проблема вимагає від студентів одночасного залучення знань із кількох дисциплінарних галузей для її цілісного осмислення. Головним завданням стає формування в здобувачів освіти навичок інтеграції різноманітної інформації, бачення зв'язків між окремими курсами та здатності синтезувати концепції з різних наук задля розв'язання комплексних практичних завдань.

PBL для трансдисциплінарного навчання розглядає знання як гнучке, контекстуальне й таке, що виходить за межі будь-яких традиційних академічних дисциплін. Навчальні проблеми створюються навколо складних, нестандартних ситуацій реального світу, де визначені дисциплінарні рамки стають перешкодою для пошуку ефективних рішень. Студенти навчаються працювати в умовах високого рівня невизначеності, комбінувати наукові теорії з особистим і професійним досвідом, а також самостійно визначати, які саме знання та підходи необхідні для вирішення проблеми, незалежно від їх галузевого походження.

PBL для критичної дискусійності (критичної компетентності) фокусується на трансформації особистості студента та розвитку критичного самоусві-

домлення. Знання у межах цього підходу розглядається не як готова істина, а як суперечливий, контекстуальний продукт, який підлягає постійному сумніву й переосмисленню. Проблеми спонукають студентів ставити під сумнів усталені догми, оцінювати альтернативні точки зору, виявляти приховані ідеологічні чи етичні підтексти та формувати власну аргументовану позицію. Освітній процес перетворюється на діалог між рівноправними учасниками, де викладач і студенти спільно досліджують складні суспільні чи професійні феномени.

На особливу увагу заслуговують позиції М. Севін-Бейден щодо трансформації розуміння проблемно зорієнтованого навчання у цифрову епоху, що представлені у дослідженні «*Rethinking Problem-based Learning for the Digital Age: A Practical Guide for Online Settings*» (2023). Насамперед наголошується на необхідності переходу від статичних текстових завдань до імерсивних, мультимедійних та нелінійних проблем, які відображають складність і невизначеність реального цифрового світу. В умовах залучення віртуальних навчальних середовищ, соціальних мереж та технологій штучного інтелекту, онлайн-формат не обмежує потенціал проблемного навчання, а навпаки – розкриває нові виміри для розвитку критичного мислення, цифрової грамотності, творчості та готовності майбутніх фахівців розв’язувати комплексні проблеми сучасності [210].

Варіативність підходів до визначення моделей проблемно зорієнтованого навчання у зарубіжному науковому дискурсі свідчить про високу гнучкість, контекстуальність та постійний еволюційний розвиток цієї педагогічної концепції – від чітко структурованої методики до багатовимірної освітньої філософії. Сучасні модифікації зумовлені специфікою різних предметних галузей, рівнем автономії здобувачів освіти, а також інституційними та культурно-освітніми традиціями конкретних країн. Також слід відзначити, що варіативність моделей пов’язана із різним розумінням сутності поняття «проблема»: від інструменту засвоєння фундаментальних знань до засобу формування критичного мислення, професійної ідентичності чи соціально-перетворювальної позиції студента.

На вибір моделі проблемно зорієнтованого навчання у вищій школі впливають такі чинники: навчальні цілі та очікувані результати (компетентності, які мають бути сформовані в процесі професійної підготовки майбутніх фахівців); контингент здобувачів вищої освіти (рівень підготовки, особливості мотивації, кількість студентів у групі); ресурси та інфраструктура у ЗВО (кадрові та матеріально-технічні ресурси, наявність часу), особливості освітньої програми, навчального плану, інституційний контекст (підтримка з боку адміністрації, готовність викладачів, рівень їхньої педагогічної підготовки).

Зарубіжні дослідники розглядають питання проблемно зорієнтованого навчання у контексті підготовки фахівців різного профілю, в тому числі й фахівців технічної галузі. Так, А. Герра (A. Guerra) відзначає, що «PBL широко визнано як підхід, придатний для формування необхідного профілю інженера 21-го століття, завдяки, наприклад, зв'язку між теорією та практикою, а також розвитку таких компетенцій, як навички вирішення проблем, комунікація та співпраця... У PBL навчальний процес починається з аналізу та формулювання проблеми з реальних і неструктурованих проблемних сценаріїв... Ці принципи дозволяють студентам розвивати високі навички міркування (наприклад, метакогнітивні знання), критичне мислення, міждисциплінарні знання, навички вирішення проблем та комунікативні навички» [188, с. 17]. Автор наголошує, що «PBL дозволяє студентам розробляти стратегії аналізу, навчання та вирішення завдань, здатні переноситися на інші контексти... PBL може підтримувати інтеграцію освіти задля сталого розвитку в інженерну освіту шляхом сприяння системному мисленню та міждисциплінарній співпраці, сприяння трансформаційному цілісному підходу до вирішення проблем та посилення зв'язків між теорією та практикою» [188, с. 17].

Проблемно зорієнтоване навчання у підготовці майбутніх інженерів розглядає М. Севін-Бейден [208]. Дослідниця стверджує, що традиційна схема «лекції + лабораторні» часто виявляється неефективною в інженерній освіті. Студенти подекуди мають серйозні труднощі щодо застосування теорії на практиці. Зокрема, аналогова електроніка вимагає не лише лінійного

математичного аналізу, а й глибокої інженерної інтуїції, яку традиційне навчання не розвиває. Для підготовки інженерів пропонуються такі інноваційні форми, як «*Problem a day*» (інтенсивна робота над однією проблемою протягом одного дня) та *Project-based problem-based learning* (поєднання PBL з реальними живими проєктами). Потенціал PBL як педагогічної основи для розвитку обчислювального мислення (*Computational Thinking*) у вищій освіті висвітлено у науковій розвідці Т. Йонасен (T. Jonasen), С. Грам-Хансен (S. Gram-Hansen) [194]. Особливо підкреслюється, що «завдяки фокусу PBL на наданні здобувачам можливості досліджувати та розвивати нові навички й розуміння у сприятливому середовищі, беручи активну участь у спільному навчанні для вирішення реальних проблем, PBL може забезпечити студентів структурою для здобуття стійких навичок обчислювального мислення, які уособлюють критичне ставлення до технологій» [194, с. 261], «обчислювальні компетенції здобувалися насамперед під час проєктної роботи та визначалися безпосередньо тією проблемою, над якою працювала група» [194, с. 263].

Багаторічний досвід розробки й успішного впровадження програми розвитку навичок розв'язання проблем (*McMaster Problem Solving Program*) на базі Університету Макмастера представлено у статті Д. Вудса (D. Woods) та його колег [223]. Відзначено, що для ефективного вирішення складних інженерних завдань недостатньо лише традиційного викладання технічної теорії. Вирішення проблем є комплексною метакогнітивною навичкою, яку необхідно викладати цілеспрямовано та системно, інтегруючи спеціальні тренінгові модулі безпосередньо у вищу технічну освіту. Автори розробили структуровану фреймворк-модель, що охоплює визначення проблеми, планування дій, моніторинг власного мислення, роботу в команді та обов'язкову рефлексію (самооцінку) після виконання завдань. Наголошується, що такий підхід значно підвищує критичне мислення студентів, їхню впевненість у власних силах і здатність вирішувати невизначені та нестандартні реальні інженерні задачі.

Світовий досвід впровадження проблемно зорієнтованого навчання в різних професійних сферах представлено у збірнику за редакцією Д. Буда

(D. Boud) та Г. Фелетті (G. Feletti) [175]. Зокрема, наголошується, що PBL – це цілісний підхід до організації освітнього процесу, який вимагає перебудови всієї освітньої програми, системи оцінювання та ролей учасників; навчання починається не з теорії чи лекції, а з комплексної, реалістичної та неструктурованої проблеми; студенти здобувають теоретичні знання саме тоді, коли відчують у них практичну потребу для розв’язання проблеми. Доцільно відзначити позицію авторів стосовно того, що традиційні тести та іспити на відтворення фактів суперечать природі PBL, тому акцент має бути на оцінюванні самого процесу розв’язання проблеми, самооцінюванні та взаємооцінюванні.

Таким чином, зарубіжні дослідники розглядають проблемно зорієнтоване навчання у вищій школі не просто як окремий методичний прийом, а як цілісну освітню парадигму, спрямовану на трансформацію студента з пасивного споживача знань у суб’єкта власного професійного зростання. В основі цього підходу лежить концепція конструктивістського навчання, де опанування теоретичним матеріалом відбувається безпосередньо через розв’язання реальних, міждисциплінарних та неоднозначних практичних ситуацій, що моделюють майбутню професійну діяльність. Ключовим елементом такого навчання є розвиток метакогнітивних навичок – здатності студентів аналізувати власний хід думок, самостійно визначати прогалини у знаннях, шукати необхідну інформацію та здійснювати критичну рефлексію. Проблемно зорієнтоване навчання передбачає кардинальну зміну ролі викладача, який із традиційного лектора перетворюється на фасилітатора, який координує групову динаміку, спрямовує дискусію та підтримує автономію здобувачів освіти. Системне впровадження технологій такого навчання підвищує мотивацію до навчання, формує критичне мислення, навички командної співпраці та забезпечує гнучкість фахівця в умовах постійних технологічних і соціальних змін.

У наукових розвідках українських дослідників позиція «проблемно зорієнтоване навчання» уперше почала використовуватися в контексті професійної підготовки майбутніх фахівців медичної сфери (М. Авраменко, О. Беляєва, Л. Горяча, Ю. Колесник, Т. Прокопенко, Л. Хламанова, Л. Хмель,

Ю. Чайковський та ін.). Автори підкреслюють, що «стратегія проблемно-орієнтованого навчання в медичній освіті передбачає наближення системи підготовки майбутніх фахівців до вимог європейського та світового рівнів, вимагає впровадження сучасних інноваційних технологій у процесі організації та здійснення освітнього процесу» [54, с. 92].

У контексті підготовки майбутніх учителів іноземних мов С. Боднар та Т. Глушко проблемно-орієнтоване навчання іншомовного усного мовлення майбутніх учителів іноземних мов визначають, як «вид навчання, що пов'язаний з: пізнавальною активністю й самостійністю студентів щодо тлумачення, дослідження й аналізу лінгвістичних явищ; сприйманням й здійсненням комунікативних дій монологічного й діалогічного характеру з орієнтацією на когнітивний розвиток студентів; формуванням в свідомості студента мовної й концептуальної картин світу крізь призму іншомовної культури; розвитком творчих здібностей та інтелектуально-креативних якостей студентів» [7, с. 102]. Автори підкреслюють, що «для проблемної ситуації важливим є акцентування на командній роботі, на відміну від завдань з розподілом на окремі групи студентів, коли кожен член групи виконує свою частину завдання, а не працює разом із командою. Для того щоб всі студенти мотивовано брали участь у розв'язування проблемної ситуації, необхідно щоб «проблема» була зрозумілою і актуальною для студентів, а також пов'язана з реальним життям чи професійною діяльністю» [7, с. 102].

Т. Семигіна зазначає, що мета проблемно зорієнтованого навчання «полягає у формуванні у студентів вміння знаходити і використовувати знання, які є актуальними і можуть бути застосовані у практичній сфері та фундаментальних дослідженнях. Окрім того, студенти вчаться ідентифікувати та аналізувати проблеми, організовувати свою діяльність із пошуку інформації, зіставляти дані з різних дисциплін, оцінювати інформацію, працювати у групі та використовувати лідерські якості» [129, с. 185].

Необхідно ще раз зазначити, що переважна більшість вітчизняних наукових публікацій з питань проблемно зорієнтованого навчання пов'язана із

професійною підготовкою майбутніх медичних працівників та учителів іноземної мови. У контексті технічної освіти та підготовки ІТ-фахівців досліджувана технологія розглядається О. Романовським та О. Лапузіною, О. Баранюк, Л. Ожиндович [120; 1; 103]. У наукових розвідках дослідників проблем професійної підготовки фахівців з комп'ютерної інженерії зазначається про використання кейс-технології/кейс-методу (Я. Сікора [136, с. 320], Л. Слободянюк [138, с. 88], К. Стрюк [145, с. 114-115], В. Круглик [62, с. 45], Л. Макаренко та Ю. Шпильовий [80, с. 82], О. Нужин [102, с. 93] та ін.

У контексті вирішення завдань модернізації вищої освіти України О. Романовський та О. Лапузіна наголошують, що в середовищі проблемно зорієнтованого навчання «студенти вивчають зміст навчальної програми в контексті проблем, з якими вони можуть стикатися в майбутньому на робочому місці. Вирішення проблеми використовується як стимул для нового навчання, а не як об'єкт для застосування раніше вивчених концепцій» [120, с. 8]. Перевага проблемно зорієнтованого навчання пов'язана з тим, що «у його основі вже лежить автентичне завдання, з яким студенти зіштовхнуться у професійному житті. Воно поділяється на безліч інших завдань, виконання яких сприятиме появі фінального результату. Наприклад, у курсах із інженерних спеціальностей чи комп'ютерних дисциплін – це може бути створення власного стартапу чи написання комп'ютерної програми, яка вирішує конкретну технічну проблему» [120, с. 10].

Відповідно до завдань підготовки майбутніх фахівців з програмної інженерії О. Баранюк підкреслює, що «проблемно-зорієнтоване навчання кардинально змінює навчальний процес і навчальні плани. Замість традиційної передачі знань від викладача до студента на перше місце ставиться проблема, яку пропонується розв'язати студентам. Проблемно-орієнтоване навчання засноване на ідеях конструктивізму, коли студент не одержує готові знання, а сам творить (конструює) свої знання шляхом активної навчальної діяльності» [1, с. 6]. У межах підготовки фахівців означеного профілю автор пропонує при викладанні курсу «Проектування

програмних систем» лабораторний цикл побудувати як серію проблем у межах одного навчального проєкту.

Використання проблемно зорієнтованого навчання у підготовці студентів технічних та ІТ-спеціальностей розглядає Л. Ожиндович, яка на прикладі курсу «Системи керування вмістом веб-сайтів» демонструє переваги методу проєктів як складника проблемно зорієнтованого навчання [103].

Визначення теоретичних засад проблемно зорієнтованого навчання зумовлює виокремлення відповідних *форм та методів* організації освітнього процесу. Зазвичай проблемне навчання або навіть проблемно зорієнтоване навчання в більшості викладачів ЗВО асоціюється із використанням методів аналізу проблемних ситуацій або роботи над кейсами. Проте Г. Барроуз у статті, опублікованій ще в 1986 р., достатньо критично висловлюється щодо таких методів, якщо вони застосовуються безпосередньо на лекціях чи практичних заняттях. Так, щодо кейсів, представлених на лекціях, відзначається, що «у кращому випадку студентів просять зрозуміти кейси, представлені з погляду інформації на лекціях, і частина цієї інформації може бути реструктуризована студентами. Може знадобитися певне висунення гіпотез, аналіз даних та обмежене прийняття рішень. Не потрібні навички дослідження чи побудови кейсів» [171, с. 483]. На лекціях, що базуються на конкретних випадках, «самостійного навчання немає, якщо студент через цікавість не шукає деякі ресурси, щоб краще зрозуміти випадки» [171, с. 483]. Щодо надання повного кейсу для вивчення та дослідження з метою підготовки до подальшого обговорення в класі відзначається, що «це більш мотивувальний метод. Однак матеріал кейсу вже організований та синтезований для студентів, що обмежує кількість міркувань, які будуть виникати» [171, с. 484].

Більш адекватним визнається проблемно зорієнтований метод на основі симуляцій, що дозволяє висувати гіпотези на основі попередніх знань, проте нова отримана інформація зазвичай не застосовується активно для переоцінки проблеми. «Замкнений цикл або повторюваний проблемний метод» передбачає оцінку студентами використаних інформаційних ресурсів,

а потім повернутися до проблеми пацієнта, як вона була представлена спочатку, щоб побачити, як вони могли б краще її обґрунтувати та отримати краще розуміння на основі того, що вони дізналися під час самостійного навчання [171, с. 484]. Важливим є зауваження Г. Барроуза про те, що проблемно зорієнтовані методи можуть бути використані і за умови самостійного навчання студентів, і при навчанні, яке спрямовує викладач. Загалом саме вивчення кейсів під наглядом тьютора залишається центральним елементом такого навчання.

У працях Г. Барроуза відзначаються також різні формати представлення проблеми – письмова справа, віньєтка справи, стандартизований (також називається одночасним) пацієнт, комп'ютерне моделювання, відеозапис [172]. У невеликих групах доцільно використовувати презентацію випадку, «дощ ідей», метод номінальних груп та консенсус щодо прийняття рішень, аналіз клінічних рішень. У контексті проблемно зорієнтованого навчання *основними рисами вдало розробленої проблеми* є такі: зв'язок з реальним життям і мотивацією студентів; опис кількох взаємопов'язаних явищ і подій; потреба у вивченні нових ключових концепцій; представлення у відповідному аутентичному контексті; має підсумковий результат, який можна застосувати в реальному житті.

У навчальних програмах проблемно зорієнтованого навчання доцільно використовувати *різні типи завдань*. Х. Шмідт (H. Schmidt) та Дж. Моуст (J. Moust) зауважують, що під час навчання студенти набувають різних видів або категорій знань про відповідні аспекти своєї галузі дослідження. Пропоновано чотири види знань: пояснювальні знання (теорії), описові знання (факти), процедурні знання (знання про те, як робити щось) та суб'єктивні знання (особисті переконання або ставлення учня). Ці види знань розглядаються в навчальних програмах, зорієнтованих на проблеми, відповідно таким типам задач: задачі-пояснення, задачі-встановлення фактів, задачі-стратегії та задачі-вирішення моральних дилем [211].

За результатами аналізу практики використання проблемно зорієнтованого навчання в Маастрихтському університеті та Києво-Могилянській академії

Т. Семигіна відзначає особливості організації освітнього процесу, у межах якого «курс розпочинається не зі вступної лекції, а одразу із семінарського (практичного) заняття з обговорення кейсу» [129, с. 187]. Відповідно, використовуються методи «мозкового штурму», дискусії. Після групової дискусії проводиться лекційне заняття, на якому наголошується на тих питаннях, які цікавили студентів, розглядаються теоретичні концепції, що стануть у нагоді при розв'язанні проблемної ситуації. Відзначається, що «при розгляді деяких кейсів лекції не проводять узагалі, у деяких випадках студенти дивляться відеоматеріали (відеолекції, відеофільми тощо)» [129, с. 187].

Для підготовки майбутніх фахівців медичної сфери в межах PLB широко використовують технологію кейсів. Наприклад, структура кейсу побудована таким чином, що студенти поступово отримують від тьютора інформацію щодо скарг, анамнезу, супутньої інформації про стан хворого, даних фізикального, інструментального та лабораторного обстеження хворого. Після отримання кожного блоку інформації студенти розпочинають дискусію між собою (це важливий момент – дискусія проводиться всередині групи студентів, тоді як викладач-тьютор виконує роль спостерігача, який мотивує студентів та направляє дискусію у випадку, коли вона непродуктивна). Основні завдання дискусії: виділити в отриманому блоці інформації вказівки, які дозволяють запідозрити ураження тієї чи тієї системи, а також містять важливу інформацію про пацієнта; скласти список питань для самостійного вивчення на підставі тих понять, термінів, даних, які студентам невідомі; скласти список робочих гіпотез [54, с. 91].

Узагальнення наукових досліджень зарубіжних та українських науковців з питань проблемно зорієнтованого навчання зумовлює висновок про такі його основні принципи:

- *проблема як відправна точка навчання* (навчання починається з автентичної, відкритої, складної проблеми, а не з подачі теоретичного матеріалу. Проблема є каталізатором, який стимулює студентів до пошуку знань. Проблеми повинні бути релевантними для майбутньої

професійної діяльності студентів або для реального світу, щоб підвищити їхню мотивацію та значущість навчання;

- *студентоцентризований підхід та автономія*. Студенти є активними учасниками освітнього процесу, беруть на себе відповідальність за власне навчання, самостійно визначають свої навчальні потреби, шукають інформацію, оцінюють її та інтегрують;
- *пріоритетна роль викладача як фасилітатора та наставника*. Викладач не надає готових відповідей, його завдання – стимулювати критичне мислення, забезпечувати ресурси, надавати конструктивний зворотний зв'язок та керувати груповою динамікою;
- *групова робота та співпраця*, що відображає реалії професійної діяльності, де співпраця є нормою, сприяє формуванню навичок комунікації, аргументування своїх позицій, активного слухання, пошуку консенсусу в обговоренні проблем, взаємонавчанню;
- *інтеграція знань та міждисциплінарність*, що зумовлено необхідністю інтегрувати знання з різних дисциплін та сфер, оскільки реальні проблеми найчастіше не обмежуються контекстом одного предмета;
- *рефлексія та метакогніція* (стимулювання процесів самоаналізу процесу вирішення проблем та власного навчання, формування здатності усвідомлювати та контролювати власне мислення та навчання).

У процесі професійної підготовки майбутніх фахівців у ЗВО проблемно зорієнтоване навчання виконує такі взаємопов'язані функції:

- *когнітивну* (спрямованість на аналіз складних ситуацій, оцінку інформації, виявлення причинно-наслідкових зв'язків, формулювання обґрунтованих висновків; усвідомлення міждисциплінарності та трансдисциплінарності сучасного знання);
- *мотиваційну* (підвищення внутрішньої мотивації здобувачів вищої освіти внаслідок роботи над автентичними, значущими проблемами; зростання залученості в контексті активного, дослідницького характеру навчальної

діяльності; створення ситуацій успіху, що сприяє підвищенню самооцінки та стимулює до навчання протягом усього життя);

- *соціально* (формування навичок співпраці та командної роботи; розвиток комунікативних навичок; умінь усного та письмового спілкування, ведення конструктивної дискусії та презентації своїх ідей; розвиток лідерських якостей, конфліктологічної компетентності);
- *професійно зорієнтовану* (підготовка до реальної професійної діяльності; формування адаптивних та преадаптивних здібностей в умовах моделювання ймовірних професійних ситуацій; формування професійних компетентностей відповідно до вимог сучасного ринку праці).

У процесі аналізу практики впровадження проблемно зорієнтованого підходу у вищій освіті науковці виокремлюють такі утруднення: доволі часто студенти, особливо перших років навчання, не готові до самостійного пошуку наукової інформації; труднощі, які виникають у викладачів під час переосмислення їхньої ролі в ході роботи зі студентами; складнощі під час формування кейсів (проблемних ситуацій) та їх постійне оновлення; формування малих груп для кращої результативності методу [178]. Т. Самигіна до утруднень при використанні проблемно зорієнтованого навчання зараховує неготовність частини викладачів до власної допоміжної ролі, не завжди вдалий добір проблемних ситуацій; споживацьке ставлення студентів, неготовність до самостійного пошуку; групова динаміка, негативні групові ролі, з якими важко було справитися студентам, які вели той чи той семінар [129, с. 188].

Основними недоліками використання проблемно зорієнтованого навчання в закладах вищої освіти Ж. Карташова, М. Кузів та О. Задоріна позначають такі: витрата часу та ресурсів – використання цього методу потребує більше часу та ресурсів як викладача, так і студентів; висока залежність процесу та результату від групи – ефективне використання методу потребує занурення у процес всіма його учасниками, а результати залежать від активності, ініціативності студентів та викладача; невідомість кінцевого результату – на відміну від традиційного навчання, де результати

оцінюються за стандартами, в PBL іноді важко передбачити результати, що може стати проблемою для викладачів; брак структури – для деяких учнів брак чіткої структури та контролю може бути проблемою, і вони можуть почуватися розгубленими; підходить не для всіх предметів – PBL може бути менш ефективним у галузях, де важливі базові теоретичні знання [43, с. 27].

Відзначені утруднення й недоліки важливі для подальшого аналізу та виявлення «вузьких місць» у практиці проблемно зорієнтованого навчання, розробки цільових стратегій та рішень для їх подолання, коригування складності кейсів і проблемних ситуацій, адаптації тієї чи тієї моделі навчання до особливостей ЗВО, уникнення поверхового або неповного засвоєння матеріалу або низького рівня розвитку компетентностей, що нівелює потенційні переваги підходу. Крім того, важливо ідентифікувати неефективне використання ресурсів та перерозподіляти їх для більшої віддачі.

Узагальнення позицій щодо проблемно зорієнтованого навчання у вищій школі представлено на Рис. 1.4.



Рис. 1.4. Концепт-карта «Проблемно зорієнтоване навчання у вищій школі»

Таким чином, сучасні вимоги до підготовки майбутніх фахівців у вищій освіті зумовлюють оновлення технологій, форм та методів організації освітнього процесу, упровадження освітніх інновацій, які сприяють формуванню навичок саморегуляції, ініціативності, що є критично важливим для навчання протягом усього життя. Проблемне навчання в контексті вітчизняної педагогічної теорії та практики зосереджується на створенні проблемних ситуацій в освітньому процесі, вирішенні учнями / студентами навчальних завдань з метою розвитку пізнавальної активності, мислення. Проблемно зорієнтоване навчання у вимірі наукових пошуків зарубіжних учених передбачає самостійну ідентифікацію проблеми, яка має комплексний характер, пов'язана із реаліями різних сфер виробничого та соціального життя, подальше міждисциплінарне дослідження, спрямоване на практичне застосування знань.

Проблемно зорієнтоване навчання не лише забезпечує засвоєння знань, але й активно розвиває комплексні когнітивні, мотиваційні, соціальні та практичні навички, необхідні для успішної реалізації в особистому та професійному житті, передбачає розвиток концептуального мислення, умінь та навичок критичного аналізу інформації, командної співпраці, формуванню комунікативних та лідерських якостей. Різні моделі проблемно зорієнтованого навчання («Pebble in the Pond» (Г. Барроуз), Challenge-Based Learning (CBL), 4C/ID, навчання на основі кейсів, проєктне навчання) сприяють активному залученню студентів в освітній процес, розвитку критичного мислення та аналітичних навичок, підготовці до реальних умов ринку праці, формуванню самостійності студентів та відповідальності за результати власної діяльності. Використання таких методів, як лекції на основі кейсів, симуляції, «повторюваний проблемний метод», аналіз кейсів, проєктне моделювання, різні формати представлення проблеми (письмова справа, віньєтка справи, комп'ютерне моделювання, відеозапис та ін.) забезпечують формування таких загальних компетентностей здобувачів вищої освіти, як критичне мислення, здатність до виявлення та вирішення проблем, уміння командної взаємодії.

1.4. Діагностика рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії

Діагностика сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії є критично важливим процесом оцінювання їхньої професійної придатності, який починається ще на етапі навчання в університеті та продовжується під час відбору технічними рекрутерами чи роботодавцями в ІТ-індустрії.

Дослідження рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії має здійснюватися на основі відповідних критеріїв та показників, що визначаються з урахуванням структурних компонентів методологічної компетентності фахівців означеного профілю, наукових підходів до діагностики пізнавальної сфери особистості. У наукових розвідках критерієм, як правило, позначають певні ознаки, що зумовлюють можливість оцінки ступеня прояву явища та процесу. Основні вимоги до критеріїв – об'єктивність, адекватність, можливість верифікації, відповідність сутнісним характеристикам досліджуваного феномена. У межах діагностичних процедур кожен критерій представлено сукупністю якісних та кількісних показників.

З урахуванням теоретичних засад дослідження, сутності та структури методологічної компетентності майбутніх фахівців з комп'ютерної інженерії визначено *критерії* сформованості досліджуваної компетентності: мотиваційний, когнітивний, праксеологічно-діяльнісний, рефлексивний та комунікативний.

Виконання завдань констатувального етапу експерименту вимагає добору цілісного комплексу діагностичних інструментів, які повинні відповідати низці обов'язкових умов: системний зв'язок із теоретико-методологічною базою дослідження, актуальність методів з погляду сучасної науки про оцінювання психолого-педагогічних явищ, а також точна відповідність віковим рамкам і соціально-психологічному профілю сучасної студентської молоді. Важливим чинником є взаємодоповнюваність окремих методик. Варто підкреслити, що у

науковому дискурсі навколо цієї проблематики дослідники застосовують різноманітний діагностичний інструментарій, зокрема анкетування, стандартизовані тести, опитувальники, карти експертного оцінювання та інші верифіковані методики. З урахуванням компетентнісного підходу в освіті вважаємо за доцільне визначити актуальність використання компетентнісно зорієнтованих завдань, кейс-тестів, які дозволяють оцінити не тільки рівень сформованості відповідних знань та умінь, а й способів діяльності, які набуті за результатами відповідної професійної підготовки, максимально наблизити процес оцінювання до реальних умов виробничої діяльності.

На підставі загальної характеристики методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, вимог до критеріальної бази у педагогічних дослідженнях дібрано комплекс відповідних методик (Таблиця 1.8).

Відзначимо, що у практиці ІТ-сфери для визначення різних аспектів професійної готовності фахівців використовуються такі методи, як автоматизоване тестування на спеціалізованих платформах, співбесіди в реальному часі, тести на проектування систем, аналіз портфоліо та аудит коду, поведінкове інтерв'ю, метод «360 градусів», кейс-тести та ситуаційні завдання. Оскільки методологічні знання та уміння є безпосередньою характеристикою пізнавальної сфери особистості, то доцільним є використання методик, спрямованих на визначення особливостей понятійного мислення, навичок аналізу та синтезу.

Таблиця 1.8

Критерії та показники сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії та діагностичний інструментарій

Критерій	Показник	Методики
Мотиваційний	Усвідомлення значущості методологічної компетентності в професійній діяльності	Анкета самооцінки (Додаток А).
	Наявність мотивації на успіх та потреби в досягненні	Методика «Мотивація до успіху» (Т.Елерс) (Додаток Б.1.).
	Інтенсивність пізнавальної потреби й інтересу до методологічної діяльності	Тест-опитувальник «Потреба в досягненнях» (Додаток Б.2).

Когнітивний	Знання загальної методології та методології інженерної діяльності	Анкета самооцінки (Додаток А) Комплекси завдань з фахових дисциплін.
	Знання щодо загальнонаукових методів пізнання та методів інженерної діяльності	
Діяльнісний	Уміння збирати інформацію з різних джерел, оцінювати її якість, достовірність	Анкета самооцінки (Додаток А), Кейс-тестінг «Самооцінка умінь та навичок роботи з інформацією» (Додаток Б.3). Методика «Дослідження узагальнення як мисленнєвої операції» (Додаток Б.4) Методика «Логічно-понятійне мислення. Утворення складних аналогій» (Додаток Б.5). Методика «Виключення понять» (здатність до класифікації та аналізу) (Додаток Б.6). Методика на визначення рефлексивності й критичності мислення (за О.Анісімовим) (Додаток Б.7). Опитувальник С.Баднера «Виявлення толерантності до невизначеності» (Додаток Б.8)
	Уміння та навички узагальнювати інформацію. Здатність до аналогії, класифікації та аналізу	
	Критичність мислення	
	Толерантність до невизначеності	
Рефлексивний	Вміння критично оцінювати результати своєї діяльності	Анкета самооцінки (Додаток А), Методика визначення індивідуальної міри вираженості рефлексивності (А.Карпов) (Додаток Б.9) Анкета «Здатність до саморозвитку» (Додаток Б.10)
	Здатність до рефлексії	
	Потреба у саморозвитку, самовдосконаленні	
Комунікативний	Здатність працювати в команді	Методика діагностики функціонально-рольових позицій у команді (Р.Белбін) (Додаток Б.11). Тест «Потреба в спілкуванні» (Ю.Орлов) (Додаток Б.12) Методика «Комунікативні й організаторські здібності» (В.Синявський, В.Федоришин) (Додаток Б.13) Кейс-тести комунікативної спрямованості (Додаток Б.14)
	Наявність комунікативних та організаторських здібностей	
	Здатність обирати адекватні методи, стилі та канали вербальної і письмової комунікації залежно від рівня технічної підготовки аудиторії.	

З урахуванням сутності компонентів методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, визначених критеріїв та показників виокремлено такі рівні сформованості досліджуваного феномену: низький, середній, високий.

Високий рівень характеризується глибоким усвідомленням стратегічної значущості методологічної компетентності для успішної професійної діяльності у динамічній ІТ-сфері. Студенти демонструють стійку мотивацію на успіх, виражену потребу в досягненнях та високу інтенсивність пізнавального інтересу до методологічної діяльності, володіють системними, фундаментальними знаннями із загальної методології та специфіки методології інженерної діяльності, чітко розуміють і вільно оперують загальнонауковими методами пізнання та спеціалізованими методами інженерії. Виявляють досконалі вміння самостійно збирати інформацію з різноманітних, зокрема англomовних та наукомістких джерел, експертно оцінювати її якість і достовірність, демонструють високий рівень навичок узагальнення, класифікації, аналізу та моделювання за аналогією. Характеризуються розвиненим критичним мисленням, високою толерантністю до невизначеності, що дозволяє ефективно розв'язувати нестандартні інженерні завдання в умовах дефіциту чітких вимог. Мають здатність об'єктивно й усебічно оцінювати результати власної праці, схильність до глибокого самоаналізу та постійну внутрішню потребу у професійному саморозвитку й самовдосконаленні. Демонструють значні лідерські та організаторські здібності, є ефективними командними гравцями й здатні гнучко адаптувати методи, стилі та канали вербальної і письмової взаємодії, легко транслуючи складні технічні концепції комп'ютерної інженерії як для колег-розробників, так і для непрофільних замовників.

Середній рівень сформованості методологічної компетентності притаманний студентам, які загалом розуміють важливість методологічної компетентності для майбутньої професії, проте їхня мотивація до успіху та пізнавальний інтерес мають переважно ситуативний характер і активізуються під впливом зовнішніх чинників або при виконанні конкретних завдань. Мають достатній обсяг знань про загальну та інженерну методологію, а також про загальнонаукові методи пізнання, проте ці знання інколи мають фрагментарний вигляд і використовуються переважно за готовими шаблонами. Здобувачі вміють знаходити інформацію та оцінювати її вірогідність за базовими критеріями,

успішно виконують операції аналізу, класифікації та узагальнення у типових ситуаціях. Критичність мислення є помірною, толерантність до невизначеності обмеженою, через що виникнення нестандартних архітектурних чи технічних проблем може викликати у них труднощі й потребувати підказок з боку керівника. Рефлексія спрямована переважно на фіксацію явних помилок, оцінка власних результатів буває поверховою, хоча прагнення до самовдосконалення присутнє у формі бажання освоїти нові прикладні інструменти комп'ютерної інженерії. У комунікативній площині є надійними виконавцями у команді, володіють базовими навичками ділового спілкування, проте під час вибору каналів і стилів комунікації їм не завжди вдається бездоганно адаптувати складну технічну термінологію під рівень підготовки слухачів, що іноді створює бар'єри у взаємодії з клієнтами.

Низький рівень виявляється у слабкому або відсутньому усвідомленні ролі методологічної підготовки в майбутній професійній діяльності. Мотивація до досягнень практично не виражена, а інтерес до методологічної діяльності зведений до мінімуму. Знання загальної методології, інженерних методів та загальнонаукового інструментарію є поверхневими, безсистемними й засвоєними на рівні механічного запам'ятовування без розуміння суті. У практичній діяльності зазнають суттєвих труднощів під час самостійного збору інформації, не вміють критично відсіювати недостовірні джерела, навички до класифікації, узагальнення не сформовані. Критичність мислення практично не виявляється, а будь-яка ситуація невизначеності чи відхилення від інструкції викликає розгубленість і нездатність продовжувати роботу над комп'ютерно-інженерним проєктом. Рефлексивні вміння майже несформовані, оскільки здобувачі не схильні аналізувати причини власних професійних невдач, перекладають відповідальність на зовнішні обставини й не виявляють явної потреби у саморозвитку. Комунікативні вміння та навички сформовані на мінімальному рівні, що проявляється в індивідуалізмі, слабких організаторських здібностях та не вмінні працювати в команді. Не здатні диференціювати стилі комунікації,

мають обмежені навички щодо грамотного та аргументованого викладу у письмовій чи вербальній формі.

Представлені рівні методологічної компетентності взаємозумовлені й перебувають у прямій залежності один від одного. Зазначені характеристики цих рівнів слугували підґрунтям для оцінювання та порівняння результатів як під час констатувального етапу експериментальної роботи, так і на етапі безпосереднього проведення педагогічного експерименту.

Необхідно підкреслити, що на початковому етапі формування методологічної компетентності здобувачів вищої освіти необхідно враховувати дискретність, нерівномірність формування досліджуваної компетентності, забезпечувати індивідуальний підхід та можливості для вибудовування індивідуальної освітньої траєкторії кожного здобувача, що дозволить ефективно компенсувати розбіжності у стартових знаннях і забезпечить умови для максимального розкриття їхнього особистісного та професійного потенціалу.

Дослідно-експериментальна робота проводилась на базі Національного університету біоресурсів і природокористування України (м. Київ) та Державного закладу „Луганський національний університет імені Тараса Шевченка” зі здобувачами вищої освіти спеціальності 123 «Комп’ютерна інженерія». Для організації експериментального дослідження було створено контрольну групу (КГ) в кількості 51 студентів та експериментальну (ЕГ) у складі 47 студентів. Такий обсяг учасників КГ та ЕГ зумовлено кількістю набору здобувачів вищої освіти на спеціальність «Комп’ютерна інженерія».

У процесі аналізу емпіричних даних під час дослідно-експериментальної роботи було дотримано відповідні вимоги: сформовані вибірки є взаємонезалежними та однорідними, а також забезпечено повну еквівалентність вихідних параметрів КГ і ЕГ (зокрема за віковим цензом, кількісним розподілом та відносно тотожним рівнем академічної успішності).

За мотиваційним критерієм рівень сформованості методологічної компетентності майбутніх фахівців з комп’ютерної інженерії визначався на підставі відповідних діагностичних методик (Табл. 1.8). За результатами

анкетування (Додаток А) було виявлено, що 74,47% ЕГ та 72,55% КГ опитаних студентів обрали спеціальність «Комп'ютерна інженерія» у зв'язку із високим рівнем заробітної плати та стабільними перспективами фінансового забезпечення в ІТ-сфері. 68,09% студентів ЕГ та 70,59% КГ зазначили інтерес до комп'ютерних технологій, бажання розібратися в архітектурі «заліза» та системного софту. Також важливою передумовою вибору професії студенти виокремили широкі можливості для працевлаштування, зокрема віддалена робота та вихід на міжнародний ринок (фріланс, аутсорс) (відповідно, 57,45% ЕГ та 62,75% КГ). Підкреслимо, що не зважаючи на загальновизнану авторитетність та престижність спеціальностей ІТ-сфери у суспільстві, такий мотив, як престижність професії та її вагома роль у розвитку сучасного цифрового суспільства, вказали замало студентів (21,28% ЕГ та 23,53% КГ). Вплив оточення (поради батьків, успішні приклади друзів чи знайомих, які вже працюють в ІТ) вказали набагато менше студентів (14,89% ЕГ, 15,69% КГ). Отже, у структурі мотивації майбутніх фахівців з комп'ютерної інженерії домінує ядро з трьох взаємопов'язаних чинників – матеріально-прагматичного (зарплата), професійно-пізнавального (інтерес до технологій) та соціально-адаптивного (міжнародний ринок, віддалена робота). Це свідчить про раціональність вибору та високий потенціал для подальшого формування професійної компетентності в цілому.

Методика «Мотивація до успіху» Т. Елерса (Додаток Б.1) спрямована на визначення рівня мотивації особистості до досягнення успіху. Важливо враховувати, що люди з помірно високим рівнем мотивації найчастіше досягають успіху, оскільки вони тверезо оцінюють ризики. А от люди із *занадто високим рівнем* часто зазнають невдач через надмірне хвилювання та схильність братися за занадто складні завдання. Опитування засвідчило, що більшість студентів ЕГ та КГ виявили середній та високий рівень мотивації до успіху (середній рівень 48,94% ЕГ та 50,98% КГ; високий рівень 36,17% ЕГ та 35,29% КГ). Такі результати, на нашу думку, пояснюються достатньою прагматичністю студентів спеціальності «Комп'ютерна інженерія», суттєву орієнтацію на кар'єрні досягнення. Низький рівень показали тільки 14,89%

студентів ЕГ та 13,73% КГ. Вважаємо за доцільне ще раз підкреслити, що фахівці ІТ-сфери мають доволі жорстку орієнтацію на результат, що підтверджується відповідями студентів на питання за шкалою Елерса щодо наполегливості та самостійності. Крім того, внаслідок високої конкуренції в ІТ-сфері студенти змушені постійно порівнювати себе із іншими та з вимогами роботодавців, що об'єктивно засвідчує їхні результати щодо достатньо високого рівня мотивації. Зазначені позиції мають бути враховані у подальшій роботі із здобувачами вищої освіти.

Суттєвим компонентом мотивації доцільно розглядати потребу у досягненні. Використання тесту-опитувальнику «Потреба у досягненні» (Додаток Б.2) засвідчило, що високий рівень потреби в досягненні виявили 44,68% студентів (ЕГ) та 43,14% студентів КГ. Середній рівень сформованості зазначеного показника зафіксовано у 34,04% досліджуваних в ЕГ та у 29,41% студентів у КГ. Низький рівень потреби в досягненні продемонстрували 21,28% майбутніх фахівців в експериментальній групі, тоді як у контрольній групі цей показник виявився суттєво вищим і склав 27,45%.

Узагальнення результатів опитування студентів щодо мотивів вибору професії, мотивації до успіху та потреб у досягненні зумовив визначення рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за мотиваційним критерієм, які представлено на Рис. 1.5. та у табл. 1.9.

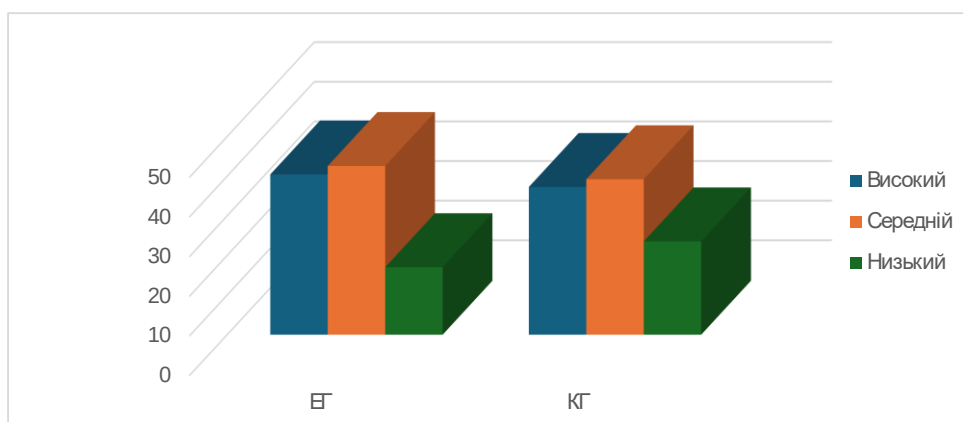


Рис. 1.5. Рівні сформованості методологічної компетентності студентів ЕГ та КГ за мотиваційним критерієм

**Рівень сформованості методологічної компетентності за мотиваційним
критерієм (констатувальний етап) (%)**

Рівні \ Групи	Експериментальна група	Контрольна група
Високий	40,43	37,25
Середній	42,55	39,22
Низький	17,02	23,53

Дослідження рівня сформованості за когнітивним критерієм здійснювалося з використання Анкети самооцінки (Додаток А), комплексу завдань з фахових дисциплін, що передбачали опору на різні аспекти використання методологічних знань студентів. Зазначена анкета зокрема містила відкрите запитання щодо розуміння студентами спеціальності «Комп'ютерна інженерія» сутності поняття «методологічна компетентність». Потенційна складність цього запитання для студентів 2 курсу полягала в тому, що ми передбачили, що студенти орієнтуються переважно на питання «технічна компетентність», використання професійних інструментів у власній діяльності. Саме такі результати очікувано й були отримані. Так, найбільш типовими відповідями були такі: *«це просто професіоналізм інженера, його знання теорії та правил розробки комп'ютерних систем»*, *«вміння працювати за правильними методами та стандартами, які придумали до нас, щоб не винаходити велосипед у комп'ютерній інженерії»*. Крім того, студенти відзначали, що *«це щось на зразок "архітектурного стилю" в голові інженера. Базові знання – це деталі комп'ютера, а методологічна компетентність – це операційна система, яка керує цими деталями та каже, як правильно їх взаємопов'язати»*, *«це як правильний пайплайн*

(черговість процесів) розробки. Вміння розкласти складне залізо чи софт на прості зрозумілі модулі й протестувати їх науковими методами».

Опитування засвідчило й суто прагматичні відповіді: *«це коли ти не просто вмієш кодувати чи збирати комп за інструкцією з Ютубу, а розумієш саму логіку: який метод підібрати, щоб система працювала швидше і без багів. Це вміння бачити всю картину проєкту перед тим, як щось робити», «для мене це як алгоритм для розв'язання будь-якої інженерної проблеми. Тобто, коли перед тобою ставлять нове завдання (наприклад, розробити мережу для офісу), ти точно знаєш, з яких кроків почати, де шукати інфу і як її аналізувати».* На жаль, але окремі студенти висловилися й так *«це коли ти вмієш писати методички для лабораторних робіт або можеш нормально пояснити матеріал іншим студентам, типу викладацькі навички»; «це коли ти знаєш напам'ять дуже багато різних методів програмування, патернів і володієш великим стеком технологій».*

Разом з тим відзначимо, що окремі студенти засвідчили й орієнтацію на дослідницький аспект, коли позначили методологічну компетентність як *«здатність швидко вчитися і розбиратися в нових технологіях. Світ комп'ютерів змінюється щодня, і ця компетентність – це вміння гуглити на професійному рівні, відсіювати фейкову інформацію від нормальної документації та застосовувати це на практиці», «вміння проводити експерименти з технікою. Наприклад, коли треба протестувати плату чи сервер під навантаженням і зробити правильні висновки, чому вони гріються або вилітають».* Як позитивне відзначимо й такі відповіді: *«це коли інженер мислить системно. Вміння декомпонувати велику задачу на дрібні і розуміти, який саме метод найкраще підійде для її вирішення на кожному етапі», «мабуть, це розуміння того, ЯК саме треба діяти. Здатність критично оцінювати різні рішення, розуміти їхні плюси і мінуси та обґрунтовувати, чому ти вибрав саме таку архітектуру бази даних чи алгоритм».*

У цілому, аналіз відповідей на зазначене питання, на нашу думку, частково пов'язані із відсутністю у навчальних планах окремих освітніх

компонентів, пов'язаних із проблемами загальної методології пізнання, методології інженерної діяльності.

У межах опитування студентів важливими були запитання щодо розуміння сутності поняття «методологія», загальнонаукових методів пізнання, співвідношення методології, методів та технології реалізації інженерних рішень. В анкеті було запропоновано закриті запитання, що уможливило подальший аналіз з урахуванням трьох рівнів (низький, середній, високий). На запитання «Як ви розумієте значення поняття «методологія» у контексті вашої майбутньої професії комп'ютерного інженера?» тільки 8,51% студентів ЕГ та 9,8% студентів КГ засвідчили розуміння методології як системи принципів і методів, що визначає стратегію інженерного мислення. Для 27,66% респондентів ЕГ та 27,45% КГ таке поняття пов'язано із інструкціями та стандартами вирішення певних питань. Ставлення до методології як суто академічного терміну з підручника показали 63,83% ЕГ та 62,75% КГ.

Студенти по різному оцінюють свої знання щодо загальнонаукових методів пізнання. Так, тільки 8,51% в ЕГ та 9,8% в КГ засвідчили своє ґрунтовне розуміння таких методів. Більшість студентів (53,19% ЕГ та 54,90% КГ) вказали на поверхові уявлення або відсутність свідомої сприйняття таких методів як елементу навчальної або майбутньої професійної роботи. Так само більшість студентів (59,57% ЕГ та 56,86% КГ) позначили сутнісні ознаки аналізу як загальнонаукового методу у зв'язку із уявним або фактичним розчленуванням об'єкта на окремі частини, а тільки 23,4% ЕГ та 23,53% КГ акцентували увагу на обов'язковому дослідженні зв'язків між елементами, визначенні причинно-наслідкових зв'язків. На жаль, 17,02% ЕГ та 19,61% КГ відповіли, що просто ознайомлення з інформацією про об'єкт та її опис загальними словами.

Достатньо інформативними були відповіді на запитання «Коли ви працюєте над проєктом, чи бачите ви різницю між методологією розробки, конкретними інженерними методами та технологіями розробки?». Більшість студентів не бачать суттєвої різниці між наведеними поняттями 63,83% ЕГ та 60,78% КГ. 27,66% студентів ЕГ та 25,49% КГ зазначили, що загалом

різницю розуміють, але у процесі реальної роботи фокусуються на технологіях та інструментах розробки. Тільки 8,51 % опитаних в ЕГ та 9,8% КГ відповіли, що чітко розмежовують ці поняття, оскільки «технології та мови (як-от SystemVerilog чи Python) змінюються, методи (наприклад, конвеєризація) є більш сталими, а методологія – це загальний концептуальний підхід, який об'єднує їх для досягнення мети».

У межах дослідження за когнітивним критерієм ми проаналізували відповіді студентів на запитання «Що таке проблема? Чи є відмінності у розумінні поняття "проблема" на побутовому, навчальному, науковому та професійному рівнях?». Наведемо декілька типових відповідей. Наприклад, Студентка П.: «Проблема – це коли є розрив між тим, що є зараз, і тим, чого хочеться досягти. У побуті це якась особиста халепа, яку зазвичай вирішуєш просто на інтуїції чи досвіді. На парах – це коли викладач спеціально підкидає нам ситуацію, типу кейс із налаштуванням мережі, щоб ми застосували те, що вивчили, і щось таки навчилися робити руками, а не тільки в теорії. А от у науці проблема – це коли є щось, чого людство ще не знає, або коли нові факти раптом не влізають у старі теорії, і доводиться копати глибше, досліджувати. Для нас, як майбутніх інженерів, проблема на професійному рівні – це коли залізо чи софт не робить того, що мало б за специфікацією. І тут вже не відбудешся інтуїцією – треба нормально розібратися, побудувати алгоритм і знайти конкретне технічне рішення». Інша відповідь студента І.: «Як на мене, проблема, це коли в системі щось йде не за планом, і треба сісти і розібратися, який тут алгоритм рішення. У побуті це взагалі якийсь хаос без структури. Ну от типовий користувач каже "у мене нічого не вантажиться" або забув пароль, і сам навіть не розуміє, чому так сталося. На парах – інша справа, там нам навмисне підкидають задачу. У науці ж проблема з'являється тоді, коли наявних алгоритмів чи потужностей заліза вже не вистачає, і доводиться шукати щось нове, досліджувати. А в професійному житті – це вже комплексне завдання. Наприклад, коли проектуєш хмарну інфраструктуру, проблема не в тому, щоб "воно просто запрацювало", а в тому, щоб продумати архітектуру як слід, налаштувати безпеку і не пересадити ресурси». Як бачимо, в принципі студенти доволі чітко розмежовують побутовий та інші виміри поняття «проблема». Однак, практично жодний студент не засвідчив такі важливі характеристики проблеми як протиріччя, наявність зовнішніх та внутрішніх чинників. Саме

тому наступна робота із здобувачами передбачала особливу увагу до навичок проблематизації, уміння формулювати проблеми в науковому та професійному середовищі.

Аналіз результатів навчальної діяльності студентів, зокрема виконання практичних завдань у межах освітніх компонентів, завдань для самостійної роботи, бесід із студентами також засвідчив обмежену поінформованість щодо сутнісних ознак загальнонаукових методів пізнання, їх можливостей у вирішенні професійних завдань.

Результати дослідження рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за когнітивним критерієм представлено в на Рис. 1.6 та у таблиці 1.10.

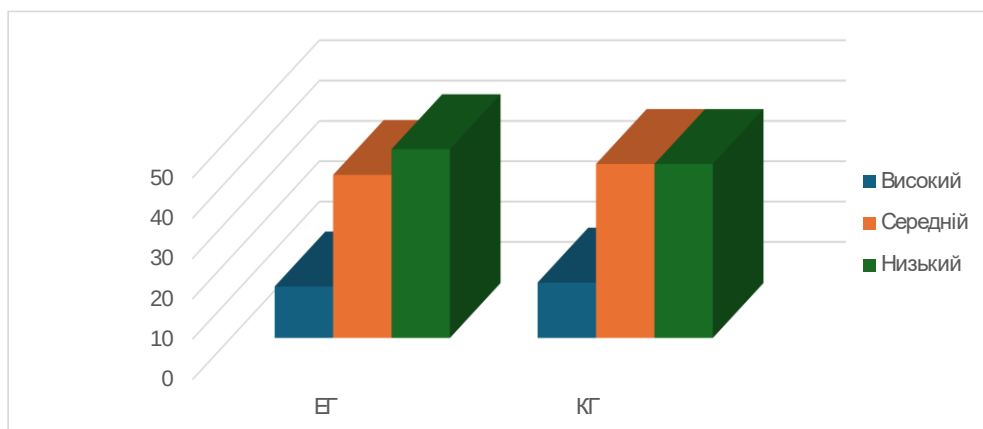


**Рис. 1.6. Рівні сформованості методологічної компетентності студентів
ЕГ та КГ за когнітивним критерієм**

Таблиця 1.10

**Рівень сформованості методологічної компетентності майбутніх фахівців
з комп'ютерної інженерії за когнітивним критерієм (%)**

Групи	Експериментальна група	Контрольна група
Рівні		
Високий	12,77	13,73
Середній	40,43	43,14
Низький	46,81	43,14

Визначення рівня сформованості досліджуваної компетентності за діяльнісним критерієм проводилося на основі спостережень та різних діагностичних методик: кейс-тестінг «Самооцінка умінь та навичок роботи з інформацією» (Додаток Б.3), методики «Дослідження узагальнення як мисленнєвої операції», «Логічно-понятійне мислення. Утворення складних аналогій», «Виключення понять» (здатність до класифікації та аналізу) (Додатки Б.4 – Б.6), методики на визначення рефлексивності та критичності мислення (Додаток Б.7), опитувальника С. Баднера «Виявлення толерантності до невизначеності» (Додаток Б.8).

Для діагностики показника «Уміння збирати інформацію з різних джерел, оцінювати її якість, достовірність» були застосовані тестові завдання, що розроблені з урахуванням контексту майбутньої професійної діяльності фахівців з комп'ютерної інженерії (авторська розробка) (Додаток Б.3). Більшість студентів виявили низький та середній рівні відповідних умінь (низький рівень у ЕГ 46.8%, у КГ – 45,1%; середній – в ЕГ 38,3%, в КГ – 39,2%). Високий рівень показали 14,9% студентів ЕГ, 15,7% – КГ. Аналіз цих результатів свідчить про виражену недостатність навичок критичної оцінки технічної інформації, оскільки здобувачі освіти здебільшого не спираються на критерії об'єктивності та достовірності. Натомість вони демонструють схильність до когнітивних упереджень, орієнтуючись виключно на власну точку зору та ігноруючи багатовимірність інженерних задач. Простежується чіткий патерн пошуку «шляху найменшого опору» – студенти масово покладаються на готові рішення, некритично імплементують згенеровані штучним інтелектом фрагменти коду чи конфігурацій, або ж задовольняються першим знайденим у пошуковиків джерелом, повністю оминаючи етап перевірки його валідності чи відповідності офіційній документації. На нашу думку, така ситуація є закономірним наслідком кількох ключових чинників, характерних для початкових етапів професійної підготовки фахівців з комп'ютерної інженерії. По-перше, в умовах різкого зростання обсягу складних фахових дисциплін (системне програмування, апаратна архітектура, мережі) студенти зазнають когнітивного перевантаження, через що їхнім пріоритетом стає отримання швидкого «працюючого» результату,

а не розуміння глибинних причин. По-друге, сучасна доступність генеративного ШІ та форумів на кшталт *StackOverflow* створює ілюзію швидкої експертності. Не маючи достатнього фундаментального досвіду, студенти не здатні розпізнати хибні кореляції або «галюцинації» нейромереж. У цілому, це свідчить про те, що традиційні репродуктивні методи викладання не створюють ситуацій когнітивного дисонансу, де б студент був змушений зіткнутися з професійним протиріччям. Це підтверджує гостру необхідність впровадження технологій проблемно зорієнтованого навчання, які створюватимуть умови для аналізу альтернативних рішень та формування критичного інженерного мислення.

Відповідно до методики «Дослідження узагальнення як мисленнєвої операції» низький рівень навичок узагальнення виявили 29,79% студентів ЕГ та 29,4% – КГ. Відповідно, середній – 51,06% та 50,98%, високий – 19,15% та 17,65 %. Представлені результати свідчать про те, що більшість студентів виявляють схильність сприймати навчальну інформацію фрагментарно та хаотично, копіювати готові практичні рішення без розуміння загальної системної логіки та концентруватися на другорядних деталях замість усвідомлення цілісної структури процесів.

Здатність до аналогій розглядалася за допомогою методики «Логічно-понятійне мислення». Утворення складних аналогій», результати якої засвідчили, що студенти відчують серйозні проблеми щодо відносин між цілим та частиною, видом та родом, потлумаченням причинно-наслідкових зв'язків. За цією методикою високий рівень виявили 17,02% студентів ЕГ та 15,69% студентів КГ, середній – відповідно 44,68% та 43,14%, низький – 38,3 та 41,18%. Маємо підкреслити, що здатність до мислення за аналогією є фундаментальним складником професійної компетентності фахівця з комп'ютерної інженерії, оскільки вона дозволяє ефективно переносити вже відомі архітектурні рішення, патерни проектування та алгоритми на розробку принципово нових технологічних систем. Завдяки цьому вмінню інженер здатний швидко адаптуватися до стрімкої зміни мов програмування чи апаратних платформ, розпізнаючи схожі логічні структури у незнайомому

середовищі та оптимізуючи процес усунення складних системних помилок. Відтак, цілеспрямована робота щодо відпрацювання таких умінь та навичок має бути включена до процесу формування методологічної компетентності фахівців зазначеного профілю.

У межах методологічної компетентності особливе значення має здатність до класифікації та аналізу. Відповідно до методики «Виключення понять» більшість студентів виявили низький та середній рівні (низький рівень – 34,04% в ЕГ та 35,29% в КГ; середній – 48,94% в ЕГ та 47,06% в КГ). Тільки 17,02% студентів ЕГ та 17,65% – КГ засвідчили високий рівень, тобто здатність чітко розмежувати суттєві й другорядні ознаки та відкидати зайве, розуміти родо-видові відносини між поняттями. Очевидно, що такі результати пов'язані із недостатнім рівнем навичок системного аналізу, схильністю спиратися на життєві чи функційні зв'язки замість логічних груп.

Критичність мислення як важливий чинник успішності реалізації завдань професійної діяльності фахівців з комп'ютерної інженерії досліджувався за методикою О. Анісімова (Додаток Б.7). У цій методиці для нас особливо цінними були аспекти роботи із проблемою, бачення протиріч. Отримано такі результати: низький рівень – 38,3% в ЕГ та 39,22% в КГ (студенти демонструють так зване «кліпове» або пасивне сприйняття інформації, схильні довіряти першим знайденим рішенням та мають труднощі з рефлексією – критичним аналізом власних помилок); середній рівень – 46,81% в ЕГ та 47,06% в КГ (студенти не виявляють системної критичності за власною ініціативою і часто приймають на віру ті чи інші припущення без їхньої попередньої перевірки); високий рівень – 14,89% в ЕГ та 13,73% в КГ (студенти мають розвинене критичне та рефлексивне мислення, здатні не просто фіксувати інформацію, а піддавати її сумніву, виявляти логічні суперечності). Відзначимо, що такі результати пояснюються не тільки індивідуальними особливостями мислення тих чи інших студентів, а й їхньою орієнтацією на швидке отримання локального результату, наприклад, написання окремого фрагмента коду за зразком, без усвідомлення його місця в загальній архітектурі. Крім того, в освітніх програмах

спостерігається очевидний брак практико зорієнтованих завдань, спрямованих на розвиток рефлексивної культури, вміння працювати з високими рівнями абстракції та критично оцінювати власні інженерні рішення.

Додатковим показником сформованості методологічної компетентності за діяльнісним критерієм обрано *толерантність до невизначеності*. Відповідно до загальновідомого опитувальника С. Баднера (Додаток Б.8) у студентів переважає середній рівень такої толерантності, що характеризується здатністю гнучко діяти в умовах дефіциту інформації лише за наявності базових інструкцій або знайомого контексту (46,81% – студенти ЕГ, 47,06% – КГ). Низький рівень толерантності до невизначеності, який свідчить про сприйняття ситуацій новизни, складності як загрози та джерела стресу, показали 38,3% студентів ЕГ та 37,25% – КГ. Відповідно, високий рівень толерантності, який відображає психологічну готовність сприймати мінливість і відсутність чітких орієнтирів як природний виклик або простір для творчості, продемонстрували лише 17,02% студентів ЕГ та 15,69% – КГ. Підкреслимо, що означений показник для фахівців з комп'ютерної інженерії є дуже важливим, оскільки сучасна ІТ-індустрія функціонує в умовах неперервних технологічних змін, розмитих технічних завдань (Legacy Code, мінливі вимоги замовника) та необхідності проектування систем, що мають працювати в непередбачуваних критичних режимах. Інженер із низькою толерантністю прагнучиме діяти виключно за жорсткими інструкціями, відчуватиме сильний стрес при виникненні нестандартних багів у коді та демонструватиме низьку адаптивність до нових мов програмування чи системних архітектур. Причини отримання незадовільних результатів на констатувальному етапі криються в академічному середовищі закладу вищої освіти, яке штучно ідеалізує освітній процес. Студентам зазвичай пропонують лабораторні та практичні роботи із заздалегідь відомим правильним результатом і чітко прописаним алгоритмом дій, де повністю відсутній елемент ризику, пошуку альтернатив чи прийняття рішень в умовах інформаційного дефіциту.

Узагальнення отриманих результатів за діяльнісним критерієм представлено на Рис. 1.7 та у Табл. 1.11.

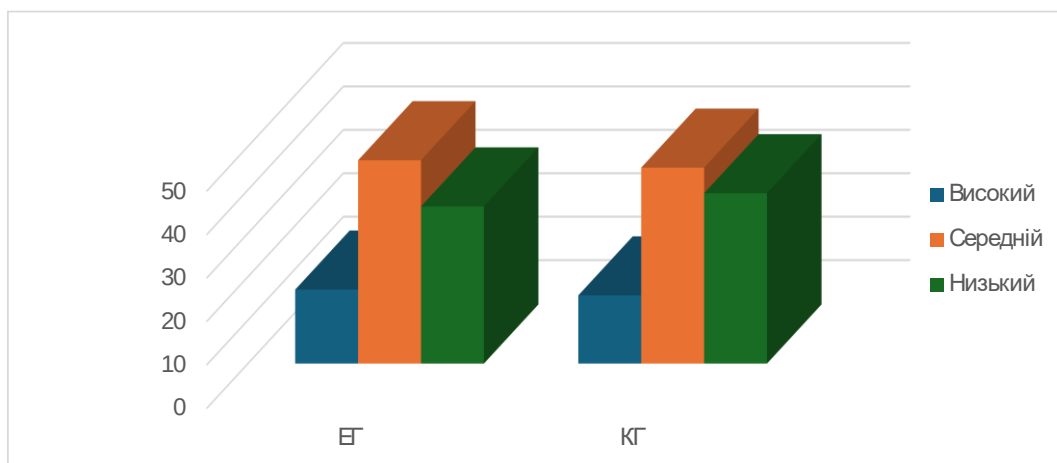


Рис. 1.7. Рівні сформованості методологічної компетентності студентів ЕГ та КГ за діяльнісним критерієм (констатувальний етап, %)

Таблиця 1.11

Рівень сформованості методологічної компетентності студентів за діяльнісним критерієм (констатувальний етап, %)

Рівні \ Групи	Експериментальна група	Контрольна група
Високий	17,02	15,69
Середній	46,81	45,10
Низький	36,17	39,22

Для діагностики за рефлексивним критерієм використано питання Анкети самооцінки (Додаток А), методику визначення індивідуальної міри вираженості рефлексивності (А. Карпов) (Додаток Б.9), анкету «Здатність до саморозвитку» (Додаток Б.10).

На запитання в анкеті «Наскільки ви внутрішньо готові до того, що через стрімкий розвиток комп'ютерних інженерних технологій вам доведеться постійно переучуватися та освоювати нові методології розробки протягом усієї кар'єри?» лише 19,15% студентів ЕГ та 17,65% – КГ відповіли, що повністю готові, оскільки динаміка змін їх надихає, а професію обрано заради постійного розвитку (високий рівень). Переважна більшість студентів (51,06% ЕГ та 50,9%

КГ) обрали варіант відповіді: «Сприймаю це як неминучу необхідність (особливого захоплення не викликає, але я готовий/-а адаптуватися, щоб залишатися на ринку праці)» (середній рівень). Значна частина респондентів (29,8% ЕГ та 31,4% КГ) зазначили варіант: «Не готовий, це мене демотивує (я хотів би один раз здобути базу знань і працювати з нею без радикальних змін)» (низький рівень).

На запитання «Чи аналізуєте ви методи, якими ви користуєтеся для навчання, пошуку інформації та вирішення інженерних завдань, з метою підвищення власної продуктивності?» отримано подібний розподіл. Лідером за кількістю відповідей виявився варіант середнього рівня: 51,06% респондентів ЕГ та 50,98% КГ вказали: «Іноді помічаю, що певні методи роботи неефективні (наприклад, витрачаю забагато часу на хаотичний пошук рішень в інтернеті), але рідко намагаюся свідомо змінити чи системно перебудувати свій підхід». Негативне ставлення та дію за звичкою («Ні, я дію за звичкою і ніколи не замислююся над тим, як я вчуся чи працюю; мене влаштовують мої поточні методи, навіть якщо вони забирають багато часу» – низький рівень) засвідчили 29,8% студентів ЕГ та 31,4% – КГ. І лише 19,1% студентів ЕГ та 17,6% КГ обрали варіант високого рівня: «Періодично аналізую свій робочий процес: помічаю, які підходи до вивчення технологій або пошуку багів працюють швидше, і намагаюся частіше використовувати саме їх».

Відповідно до методики А. Карпова низький рівень рефлексивності, який вказує на недостатній розвиток самоаналізу, труднощі у прогнозуванні наслідків власних дій та слабку схильність до критичного переосмислення минулого досвіду, зафіксовано у 36,2% студентів ЕГ та у 35,3% представників КГ. Середній рівень, що відображає здатність до ситуативної рефлексії, коли аналіз власних мисленнєвих операцій та інженерних рішень активізується переважно під зовнішнім контролем викладача або у разі виникнення явних збоїв у роботі системи, виявлено у 46,8% в ЕГ та 47,1% в КГ. Високий рівень рефлексивності, який свідчить про розвинену здатність до постійного самоконтролю, глибокого аналізу системних помилок та усвідомленого керування процесом власного

навчання і проєктування, виявлено тільки в 17,02% респондентів ЕГ та 17,65% студентів КГ. На нашу думку, значна частка студентів із низьким та середнім рівнями рефлексивності зумовлена специфікою організації традиційного навчання, де акцент зазвичай робиться на результативності виконання інженерного завдання, а не на осмисленні самого процесу його розв'язання. Відтак, майбутні фахівці майже не залучаються до метакогнітивної діяльності, тобто до аналізу власних мисленнєвих операцій, виявлення внутрішніх дефіцитів, що призвели до певних помилок. І саме це може суттєво утруднювати формування методологічної компетентності.

У межах дослідження за показниками рефлексивного критерію використано анкету «Здатність до саморозвитку» (Додаток Б.10). Низький рівень, який за цією анкетною інтерпретується як «зупинка розвитку», виявлено в 25,5% студентів ЕГ та у 27,45% студентів КГ. Середній рівень, що свідчить про відсутність стійкої системи саморозвитку та сильну залежність орієнтації на розвиток від зовнішніх умов і стимулів показали 53,2% респондентів ЕГ та 52,9% – КГ. Високий рівень, що відповідає стадії «активного розвитку», та притаманний студентам із чіткою внутрішньою мотивацією до неперервного професійного зростання, засвідчено в 21,28% респондентів ЕГ та 19,61% – КГ. Такі результати свідчать, що більшість сучасних студентів орієнтовані на саморозвиток лише тоді, коли цього безпосередньо вимагає навчальний контекст (іспити, заліки чи дедлайни проєктів), або у разі виникнення прямої прагматичної вигоди, наприклад, перспективи працевлаштування. Разом з тим, внутрішня система самокорекції, читання фахової літератури та глибока рефлексія власних почуттів і досвіду поза університетською програмою залишаються несформованими, що закономірно актуалізує потребу в додатковій цілеспрямованій роботі у межах освітнього процесу для стимулювання переходу студентів до усвідомленого й активного саморозвитку.

Узагальнені результати за рефлексивним критерієм представлено на Рис. 1.8 та у Табл. 1.12.

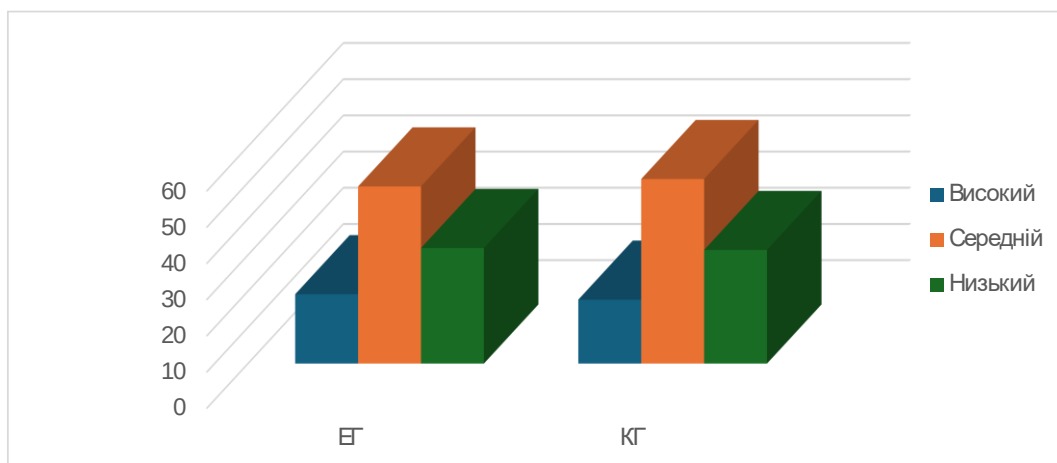


**Рис. 1.8. Рівні сформованості методологічної компетентності студентів
ЕГ та КГ за рефлексивним критерієм (констатувальний етап, %)**

Таблиця 1.12

**Рівень сформованості методологічної компетентності майбутніх фахівців
з комп'ютерної інженерії за рефлексивним критерієм
(констатувальний етап, %)**

Групи	Експериментальна група	Контрольна група
Рівні		
Високий	19,15	17,65
Середній	48,94	50,98
Низький	31,91	31,37

Оскільки в сучасних умовах розвитку ІТ-сфери поцінуються не тільки знання та уміння з фахових дисциплін, але й наявність певних soft skills, серед критеріїв сформованості методологічної компетентності виокремлено й комунікативний. Окремі питання в Анкеті самооцінки (Додаток А) були спрямовані на самооцінку комунікативних навичок в професійному середовищі. Аналіз відповідей студентів на запитання щодо володіння професійною термінологією з комп'ютерної інженерії під час обговорення проєктів засвідчив однорідність вихідних даних ЕГ та КГ. Зокрема, більшість респондентів продемонстрували середній рівень (48,94%

в ЕГ та 49,02% у КГ), зазначивши, що при намаганні спілкуватися фаховою мовою вони іноді плутають специфічні терміни чи стандарти. Близько третини студентів виявили низький рівень (34,04% в ЕГ та 33,33% у КГ), замінюючи точні поняття побутовими лексемами. Високий рівень володіння термінологією та здатність вести професійний діалог «на рівних» зафіксовано лише у 17,92% студентів ЕГ та 17,65% КГ. Аналіз самооцінки студентів щодо здатності презентувати та захищати розроблені технічні рішення зумовив висновок про практично однаковий вихідний стан ЕГ та КГ. Більшість респондентів виявили середній рівень (48,9% в ЕГ та 49,02% у КГ), зазначивши, що можуть виступати лише за стандартним шаблоном і відчують труднощі при необхідності аргументувати вибір альтернатив. Близько третини студентів (34,04% в ЕГ та 33,3% у КГ) продемонстрували низький рівень, намагаючись уникати публічних захистів через складність у структуруванні технічної розповіді. Високий рівень (здатність переконливо аргументувати розробки та впевнено відповідати на критику) зафіксовано лише у 17,02% студентів ЕГ та 17,6% КГ.

Комунікативні та організаційні якості, уміння працювати в команді на сьогодні розглядаються значущими в професійній компетентності фахівців. Орієнтацію на певні функційно-рольові позиції в команді дозволяє дослідити відома методика Р. Белбіна (Додаток Б.11). Для успішності інженерної команди особливо значущими є такі ролі: IV «Оцінювач ідей / Аналітик», V – «Організатор роботи / Спеціаліст», IX – «Завершувач / Фіналізатор», I «Мислитель / Інноватор» (високий рівень), VI – «Координатор / Якір», II – «Формувальник / Мотиватор», VIII – «Запроваджувач / Піонер» (середній рівень). Низький рівень командної роботи, як правило, пов'язано із ролями, що фокусуються на соціальному аспекті, корисні для здорової атмосфери, але за умови їхнього домінування виникають проблеми щодо виконання суто технічних завдань (VII – «Організатор команди / Помічник», III – «Генератор ідей / Нетворкер»). Аналіз опитування студентів засвідчив наявність високого рівня тільки у 17,02% студентів ЕГ та 15,7% – КГ, що пов'язано із

недостатнім практичним досвідом та навичок аналітичної роботи. Середній рівень виявлено у 46,8% студентів ЕГ та 47,1% – КГ. Більшість студентів вже здатні виступати виконавцями чутко структурованих завдань під керівництвом лідерів, шукати готові технічні ідеї та координувати поточні процеси всередині мікрогруп. Показники низького рівня виявлено у 36,2% студентів ЕГ та 37,2% – КГ, що відображає прагнення компенсувати відсутність міцних технічних знань суто соціальною взаємодією або хаотичним генеруванням ідей.

Відповідно до тесту «Потреба в спілкуванні» (Додаток Б.12) отримано такі результати: низький рівень – 36,2% в студентів ЕГ, 35,3% – в студентів КГ, середній – 48,9% в ЕГ та 51,0% в КГ, високий – 14,9% (ЕГ), 13,7% (КГ). У цілому більшість студентів демонструють збалансовану комунікабельність, поєднання прагнення до зосередженої індивідуальної роботи з необхідністю підтримувати стабільні контакти та орієнтацію на сумісну роботу. Однак, відзначимо, що незважаючи на особливості сучасного покоління молоді як цифрових комунікаторів значна частка студентів тяжіють до інтровертивного типу поведінки, свідомо обирають індивідуалізовані форми роботи.

Аналіз комунікативних та організаторських здібностей (Додаток Б.13) зумовлює такі висновки: низький рівень – 21,3% (ЕГ), 21,6 (КГ), середній – 53,2% (ЕГ), 54,9% (КГ), високий – 25,5% (ЕГ), 23,5% (КГ). Підкреслимо, що специфіка сучасної ІТ-сфери вимагає навичок комунікації у межах командної роботи, що актуалізує питання залучення майбутніх фахівців до інтерактивних форм роботи та стимулювання навичок вербальної та писемної комунікації.

Додатковим інструментом дослідження за комунікативним критерієм слугували завдання у межах кейс-тестінгу (Додаток Б.14). Наведемо приклад типового завдання, яке відповідає вимогам до компетентнісних завдань у вищій школі: *«Кейс-тест 1. Вам потрібно повідомити групі бухгалтерів (нетехнічних користувачів), що на сервері проводяться планові технічні роботи, через які база даних буде недоступна 2 години. Який варіант повідомлення є найбільш адекватним? А. База даних тимчасово відключена*

через критичні помилки в архітектурі. Можливо, деякі дані будуть втрачені під час рестарту. Б. Шановні колеги! З 14:00 до 16:00 ми проводимо оновлення системи для її стабільної роботи. Протягом цього часу програма обліку буде недоступна. Просимо заздалегідь зберегти документи. В. Сервер 192.168.1.10 буде в стані 'down' з 14:00. Всі запити по порту 1433 будуть ігноруватися. Г. Ми проводимо реіндексацію SQL-таблиць та оновлення патчів безпеки ядра ОС. Система повернеться до режиму 'online' після завершення деплою». Виконання подібних кейс-тестів вимагає насамперед розуміння специфіки аудиторії, з якою комунікує фахівець та вибору адекватних форм повідомлень. За результатами виконання студентами кейс-тестів щодо комунікації отримано такі результати: низький рівень – 42,5% (ЕГ), 43,1% (КГ); середній рівень – 46,8% (ЕГ), 45,1% (КГ); високий рівень – 10,6% (ЕГ), 11,76% (КГ). Зазначимо, що для окремих студентів виконання таких завдань викликало певне здивування, оскільки такі питання не розглядалися ними як важливі для майбутньої професійної діяльності.

Результати дослідження за комунікативним критерієм представлено на Рис. 1.9 та у Таблиці 1.13.

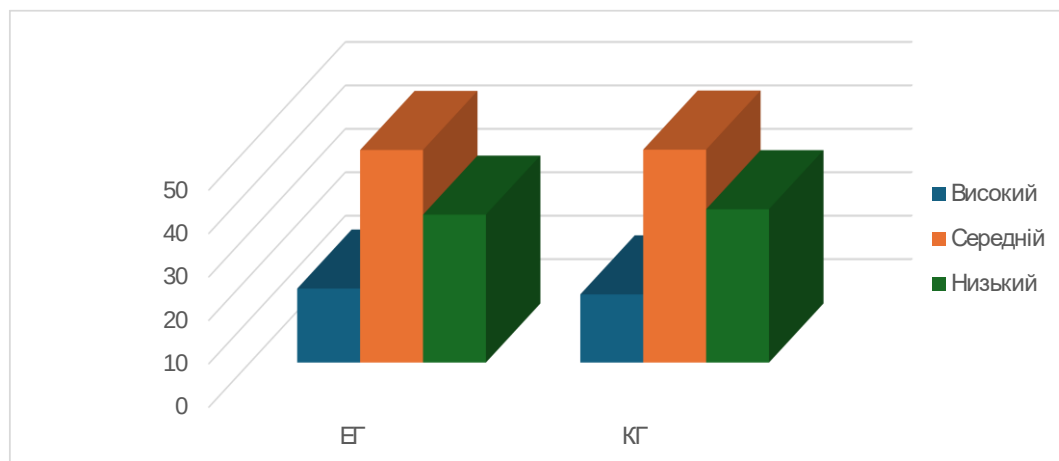


Рис. 1.9. Рівні сформованості методологічної компетентності студентів ЕГ та КГ за комунікативним критерієм (констатувальний етап, %)

**Рівень сформованості методологічної компетентності майбутніх фахівців
з комп'ютерної інженерії за комунікативним критерієм
(констатувальний етап, %)**

Рівні \ Групи	Експериментальна	Контрольна
	група	група
Високий	17,02	15,69
Середній	48,94	49,02
Низький	34,04	35,29

Діагностичні методи та методики, що використовувалися на констатувальному етапі дослідження, дозволили здійснити аналіз рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за відповідними критеріями та показниками. У цілому рівень сформованості досліджуваної компетентності визначаємо як переважно середній. Таким чином, наявність суперечностей між технологічною складністю сучасних обчислювальних систем, високими вимогами ІТ-індустрії до системного мислення і недостатньою увагою до методологічної підготовки у процесі фахового становлення та розвитку призводять до формування фахівців, які мають певний набір технічних знань і навичок, однак не здатні до цілеспрямованого використання загальнонаукових методів, реалізації вимог загальної методології та методології комп'ютерної інженерії у професійній діяльності. Результати констатувального етапу свідчать про недостатню увагу до методологічного складника підготовки ІТ-фахівців у ЗВО, що потребує розробки та впровадження педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, зокрема засобами технологій проблемно зорієнтованого навчання.

ВИСНОВКИ ДО РОЗДІЛУ 1

Стрімка зміна архітектурних рішень та мов програмування в ІТ-індустрії вимагає від майбутнього інженера не просто володіння поточними інструментами, а глибокого методологічного мислення, системного бачення розробки та здатності до самостійного моделювання складних комп'ютерних систем. Гостра потреба у формуванні методологічної компетентності актуалізується також інтенсивним впровадженням технологій штучного інтелекту та кіберфізичних систем, де звичайний репродуктивний підхід до проєктування поступається місцем наукомістким та крос-доменним інженерним рішенням. Відтак, без належного методологічного фундаменту випускники технічних університетів ризикують втратити конкурентоспроможність на ринку праці вже через кілька років після закінчення навчання через швидке моральне застарівання вузькоприкладних знань.

Аналіз дотичних до поняття «методологічна компетентність» понять («методологія», «метод», «методологія практичної діяльності») зумовив висновки про те, що знання методології та її практичних засад є тим універсальним компасом, який допомагає кожній людині хаотичний потік інформації перетворювати на чіткі, послідовні та ефективні дії. Незалежно від сфери – побудова кар'єри, вирішення побутових завдань чи особистісний розвиток – розуміння правильних методів і підходів захищає від критичних помилок і дозволяє свідомо досягати поставлених цілей.

Дослідження засвідчило, що методологічна компетентність фахівців різного профілю розглядається як складна інтегративна характеристика особистості фахівця, що виражається у володінні теорією і практикою застосування методів пізнання та перетворення дійсності, здатності до системного проєктування професійної діяльності, ціннісному ставленні до вибору інструментарію та постійній рефлексивній готовності до оновлення методологічних засад праці в умовах невизначеності.

Методологічну компетентність майбутнього фахівця з комп'ютерної інженерії було визначено на основі сутності понять «методологія», «компетентність», «методологічна компетентність», вимог сучасної ІТ-сфери до професійної діяльності таких фахівців як компонент професійної компетентності, динамічне інтегративне особистісне утворення, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проектування для розв'язання складних технічних задач. Така компетентність виступає регулятивом професійної діяльності, що дозволяє фахівцям не лише адаптуватися до швидкої зміни інструментарію, а й самостійно конструювати нові алгоритми розв'язання складних науково-технічних задач на основі системного аналізу.

На основі аналізу наукових джерел, нормативних документів схарактеризовано особливості професійної підготовки майбутніх фахівців з комп'ютерної інженерії: інтеграція апаратного та програмного складників у змісті навчання як відповідь на стрімкий розвиток інтернет-речей, штучного інтелекту та хмарних обчислень; перехід від класичного вивчення архітектури обчислювальних систем до практико зорієнтованого проектування складних кіберфізичних комплексів; необхідність формування високого рівня адаптивності та навичок неперервного самонавчання через надзвичайно короткий життєвий цикл інновацій у галузі.

На основі аналізу наукових джерел та визначених функцій (когнітивна, нормативно-регулятивна, проектно-конструктивна, рефлексивна, прогностична, преадаптивна та комунікативна) виокремлено такі структурні компоненти методологічної компетентності фахівців з комп'ютерної інженерії, як мотиваційний, когнітивний, праксеологічно-технологічний, рефлексивний та комунікативний.

Аналіз феномена методологічної компетентності та реалізація завдань формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії увиразнює низку взаємопов'язаних та

комплементарних методологічних підходів: системний, діяльнісний, особистісно зорієнтований, компетентнісний та праксеологічний.

Упровадження проблемно зорієнтованого навчання у вищій школі є критично важливим кроком для подолання розриву між теоретичною підготовкою та мінливими вимогами сучасного ринку праці, оскільки цей підхід трансформує студента з пасивного слухача на активного дослідника, здатного системно мислити та ухвалювати виважені рішення в умовах невизначеності. Доведено, що найбільш актуальними моделями такої освітньої парадигми на сьогодні є класична медична модель Г. Бароуза, інтегровані міждисциплінарні та гібридні цифрові моделі, які успішно реалізуються через технології кейс-методу, командно зорієнтованого навчання, проєктно-дослідних технологій та імітаційного моделювання реальних професійних ситуацій за допомогою штучного інтелекту. Ефективне розгортання цих технологій забезпечується взаємодоповненням таких методів, як евристична бесіда, метод «мозкового штурму», дискусійний аналіз дилем та контекстне моделювання виробничих завдань, що у своїй сукупності не лише гарантує глибоке засвоєння фахових знань, а й формує ключові гнучкі навички, інноваційну культуру та готовність майбутніх спеціалістів до неперервного навчання впродовж усього життя.

З урахуванням теоретичних засад дослідження визначено такі критерії сформованості у майбутніх фахівців з комп'ютерної інженерії методологічної компетентності, як мотиваційний, когнітивний, діяльнісний, рефлексивний та комунікативний з відповідними показниками та їх рівні – низький, середній, високий.

З метою діагностики та виявлення рівнів сформованості досліджуваної компетентності було дібрано комплекс відповідних методик: «Мотивація до успіху» (Т. Елерс), тест-опитувальник «Потреба в досягненнях», кейс-тестінг «Самооцінка умінь та навичок роботи з інформацією», методики «Дослідження узагальнення як мисленнєвої операції», «Логічно-понятійне мислення. Утворення складних аналогій», «Виключення понять» (здатність

до класифікації та аналізу), опитувальник С. Баднера «Виявлення толерантності до невизначеності», методика діагностики функційно-рольових позицій у команді (Р.Белбін) та ін.

Аналіз даних констатувального етапу дослідження зумовив висновок, що рівень сформованості методологічної компетентності майбутніх фахівців з комп'ютерно інженерії недостатній, відсутнє комплексне розуміння значущості загальної методології, методології інженерної діяльності, загальнонаукових методів пізнання у вирішенні проблем навчальної та майбутньої професійної діяльності.

Отримані результати зумовили висновок про необхідність цілеспрямованого формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованого навчання.

РОЗДІЛ II

ЕКСПЕРИМЕНТАЛЬНЕ ДОСЛІДЖЕННЯ ПЕДАГОГІЧНИХ УМОВ ФОРМУВАННЯ МЕТОДОЛОГІЧНОЇ КОМПЕТЕНТНОСТІ МАЙБУТНІХ ФАХІВЦІВ З КОМП'ЮТЕРНОЇ ІНЖЕНЕРІЇ ЗАСОБАМИ ТЕХНОЛОГІЙ ПРОБЛЕМНО ЗОРІЄНТОВАНОГО НАВЧАННЯ

2.1. Обґрунтування педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання

Сучасні вимоги до професійної діяльності майбутніх фахівців з комп'ютерної інженерії визначаються стрімким розвитком високих технологій, що зміщує акцент з виконання рутинних операцій на здатність проєктувати та супроводжувати складні інтегровані системи. Спеціаліст нової генерації має гнучко поєднувати глибокі фундаментальні знання архітектури обчислювальних систем та системного програмування з навичками розробки інтелектуальних пристроїв, адміністрування хмарних сервісів та забезпечення кібербезпеки на всіх рівнях інфраструктури. Окрім суто технічної експертизи, критично важливими стають готовність до постійного самонавчання в умовах швидкої зміни інструментарію, здатність ефективно працювати у крос-функційних командах за гнучкими методологіями, а також уміння аналізувати економічну доцільність та екологічність інженерних рішень. Таким чином, від випускника вимагається не просто володіння актуальними програмно-апаратними засобами, а сформованість цілісного інженерного мислення, методологічної компетентності, що дозволяє самостійно розв'язувати нестандартні технічні завдання та адаптуватися до динамічних запитів глобального ринку праці.

Ефективне формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії характеризується наявністю системності і можливе лише за певних педагогічних умов.

Сутнісні характеристики поняття «педагогічні умови» репрезентовано у наукових розвідках А. Литвина [73], Н. Тверезовської та Л. Філіппової [148], Є. Хрикова [154].

У тлумачному словнику української мови поняття умова потрактовується, як «1) необхідна обставина, яка робить можливим здійснення, створення, утворення чого-небудь або сприяє чомусь, 2) обставини, особливості реальної дійсності, при яких відбувається або здійснюється що-небудь, 3) правила, які існують або встановлені в тій чи іншій галузі життя, діяльності, які забезпечують нормальну роботу чого-небудь; правила, вимоги, виконання яких забезпечує що-небудь, 4) сукупність даних, положення, що лежать в основі чого-небудь» [101, с. 617]. Педагогічні умови у педагогічному словнику представлено, як «обставини, від яких залежить та відбувається цілісний продуктивний педагогічний процес професійної підготовки фахівців, що опосередковується активністю особистості, групою людей» [140, с. 243].

У психолого-педагогічних джерелах педагогічні умови класифікують за різними ключовими ознаками. Так, залежно від середовища їх розмежовують на екзогенні / зовнішні (природно-кліматичні, суспільні, виробничі, культурно-освітні тощо) та ендогенні / внутрішні (матеріально-технічні, морально-етичні, естетичні й подібні); за особливостями впливу – на об’єктивні (законодавча та нормативна база, мас-медіа, динамічне середовище, що підтримує діяльність педагогічної системи та стимулює належну поведінку її учасників) й суб’єктивні (чинники, що визначають особливості освітнього середовища, зокрема потенціал учасників навчання, злагожденість їхньої співпраці, ступінь індивідуальної цінності освітніх цілей для здобувачів); за особливостями учасників комунікації – загальні (суспільні, фінансові, культурні, етнічні) та специфічні (демографічний та соціальний аспекти навчальних колективів, розташування освітньої установи, її ресурсне забезпечення, рівень професійної культури викладачів, а також морально-психологічний клімат в студентському чи учнівському середовищі).

Науковці виокремлюють різні варіанти педагогічних умов. Так, позначаються організаційно-педагогічні, психолого-педагогічні, дидактичні тощо умови (Н. Тверезовська, Л. Філіппова та ін.). Організаційно-педагогічні умови, з одного боку, включають комплекс можливостей для ефективного розв'язання освітніх завдань (поєднання змістових елементів, форм і методів цілісного педагогічного процесу), а з іншого – передбачають керованість освітньої системи та її орієнтацію на подальший функційний розвиток (сукупність об'єктивних чинників, матеріально-технічне забезпечення, організаційні форми, методи і прийоми для цілеспрямованої співпраці суб'єктів педагогічної взаємодії). Психолого-педагогічні умови охоплюють чинники, які забезпечують безпосередній педагогічний вплив на суб'єктів чи об'єктів освітньої системи, спрямований на розвиток, виховання та навчання особистості. Дидактичні умови розглядаються як комплекс чинників, провідна функція яких полягає у цілеспрямованому виборі та реалізації можливостей змісту, методів, форм і засобів педагогічної взаємодії в освітньому процесі.

Визначення педагогічних умов зумовлено специфікою освітнього процесу, структурою особистісних якостей, що підлягають трансформації під впливом педагогічних засобів, метою виокремлення, проектування та застосування складників змісту, методів, прийомів та організаційних моделей навчання.

Для обґрунтування педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії доцільно звернутися до наукових розвідок, в яких представлено педагогічні умови, технологічне забезпечення формування означеної компетентності, а також методологічної культури та дослідницької компетентності фахівців різного профілю.

У контексті підготовки майбутніх учителів музичного мистецтва Лі Ліцюань зазначає такі педагогічні умови: «активізація пізнавальної мотивації студентів на опанування основами наукового пізнання вокального виконавства й вокальної педагогіки; застосування наукової екстраполяції як методологічної процедури в процесі вокальної підготовки майбутніх

учителів музичного мистецтва; включення студентів у процесі вокальної підготовки до різних видів науково-дослідницької діяльності; стимулювання розвитку науково-методичної творчості майбутніх учителів музичного мистецтва на заняттях з вокальних дисциплін для досягнення їх наукового, професійного та особистісного акме» [75, с.112]. Автор підкреслює доцільність таких методів, як евристична бесіда, пошукові, проблемно-пошукові та творчо-аналітичні завдання, науково-методичні проєкти, методи семіотичного та герменевтичного аналізу та ін. [75, с. 117-128].

О. Савицька відзначає зовнішні та внутрішні умови формування методологічної компетентності студентів [122, с. 73-74]. Зовнішні: а) система наукової комунікації, яка функціонує у закладі вищої освіти... залучення студентів до науково-дослідницької діяльності; б) структурно-динамічні особливості освітнього простору, в якому на різних рівнях розгортається вся система педагогічних та соціально-психологічних взаємодій між суб'єктами освітнього процесу; в) особистість наукового керівника. Досить часто саме від науковця залежить розвиток у студента релевантної мотивації наукового пізнання, його інтелектуальних властивостей, вольових якостей та креативності. Внутрішні: а) пізнавальна спрямованість студента, яка забезпечує вибірковість ставлення особистості до різних об'єктів пізнавальної, навчальної та дослідницько-пошукової діяльності та визначає специфіку протікання і результативність цих видів діяльності; б) дослідницькі здібності студентів як індивідуально-психологічні особливості особистості, що забезпечують успішне здійснення наукової діяльності. Дослідницькі здібності включає такі компоненти: а) особистісний; б) когнітивний; в) емоційно-вольовий; г) діяльнісний.

Значущість «практично-орієнтованих форм, методів та освітніх технологій на засадах гуманістичних принципів і підходів, зокрема емпатії, толерантності, рівності освітніх можливостей, мобільності й динамічності в реагуванні на сучасні соціально-економічні виклики та потреби» у процесі формування методологічної компетентності аспірантів відзначає Л. Нічуговська [100, с. 41].

Педагогічними умовами формування методологічної культури майбутніх учителів природничих дисциплін О. Лаврентьєва зазначає такі позиції: «готовність викладачів до вироблення в студентів специфічного для сфери освіти стилю мислення, посилення наскрізної загальної та спеціальної методологічної підготовки, використання механізмів методологічної рефлексії у сприйнятті, розумінні, проєктуванні та конструюванні майбутніми вчителями педагогічної реальності, педагогічний моніторинг стану педагогічної системи розвитку методологічної культури майбутнього вчителя природничих дисциплін» [70, с. 269]. Автором наголошено на значущості проблемних ситуацій, евристичних прийомів вирішення та моделювання педагогічних ситуацій, задач алгоритмічного типу, навчальних задач, побудованих на контекстній основі, моделювання педагогічних ситуацій, кейс-методу, проєктних технологій, ігрового проєктування, тренінгові технології [70, с. 356-374].

У контексті формування науково-дослідницької компетентності майбутніх інженерів-програмістів М. Вінник відзначає такі умови: «1) створення у вишій науково-освітнього професійного середовища; 2) освітня науково-дослідницька інтеграція викладачів, провідних ІТ-фахівців та студентів; 3) стимулювання мотивації студентів до науково-дослідницької діяльності; 4) поетапне залучення студентів до науково-дослідницької діяльності» [17, с. 129-130]. Л. Бондаренко виокремлює умови формування дослідницької компетентності майбутніх викладачів: «розвиток дослідницького потенціалу студентів магістрантів на основі актуалізації внутрішніх ресурсів особистості в процесі оволодіння фаховими дисциплінами; стимулювання магістрантів до науково-дослідницької діяльності шляхом залученням їх до розв'язання випереджувальних пізнавальних та дослідницько-творчих завдань професійно зорієнтованого характеру під час застосування різноманітних форм і методів організації освітнього процесу; включення магістрантів у різні види науково-дослідної роботи в навчальному процесі, у ході різних видів практики, самостійної дослідницької діяльності в позанавчальний час» [8, с. 94]. Для забезпечення формування дослідницької компетентності майбутніх спеціалістів з економічної кібернетики М. Фролова

пропонує такі умови: «упровадження особистісного підходу до формування дослідницьких компетентностей майбутніх спеціалістів з економічної кібернетики; ознайомлення майбутніх спеціалістів з економічної кібернетики з різновидами знань про дослідницьку діяльність; оволодіння майбутніми спеціалістами з економічної кібернетики способами дослідницької активності; системне використання освітньо-виховних можливостей фахової підготовки у формуванні дослідницьких компетентностей майбутніх спеціалістів з економічної кібернетики» [158, с. 130-131].

У контексті розвитку дослідницької компетентності магістрів спеціальності інженерія програмного забезпечення Н. Котенко акцентує увагу на значущості таких форм та методів, як залучення до науково-дослідної роботи, діяльності дослідницьких гуртків, участі у конференціях та вебінарах, студентських конкурсах, виконання практико орієнтованих завдань [60, с. 611-612]. Формування дослідницької компетентності майбутніх докторів філософії, як зазначає І. Олійник, передбачає використання таких методів організації освітньої діяльності, як «рефлексивні методи, що базуються на індивідуальних якостях, самоаналізі, усвідомленні власних ресурсів, задатків, умінь (самооцінка, самоаналіз тощо); тренінгово-ігрові методи та технології, що забезпечують інтенсивне освоєння індивідуального та групового досвіду, корекцію знань, професійних умінь і навичок, формування необхідних компетентностей в спеціально створених умовах (ділові ігри, тренінги тощо); інноваційно-діяльнісні методи, які ґрунтуються на новітніх освітніх технологіях (алгоритмізація, моделювання, проєктування тощо)» [104, с. 312].

Узагальнення наукових підходів сучасних дослідників до виокремлення педагогічних умов формування методологічної компетентності, дослідницької компетентності фахівців різного профілю, форм та методів реалізації такого процесу зумовлює висновок, про те, попри специфіку кожної професійної сфери, у поглядах дослідників чітко простежується інваріантна (спільна) структура, яка охоплює *чотири ключові напрями виокремлення педагогічних умов* – орієнтація на забезпечення формування мотиваційно-ціннісного компонента, наскрізна

інтеграція науково-дослідницької діяльності в освітній процес, використання інтерактивних та практико зорієнтованих технологій навчання, етапність та моніторинг результатів. *Провідне місце в освітньому процесі* посідають пошуково-евристичні методи, кейс-технології, ігрове проектування, а також алгоритмізація та моделювання реальних професійних ситуацій. Розвиток методологічної рефлексії та самоаналізу забезпечується через тренінгово-ігрові форми взаємодії, ділові ігри, коучинг та фасилітацію. Важливим є також залучення здобувачів до роботи наукових шкіл, дослідницьких гуртків, майстер-класів, а також через їхню активну участь у конференціях, вебінарах і конкурсах проєктів.

Визначення теоретико-методичних засад дослідження, аналіз результатів констатувального етапу експерименту, наукових положень щодо поняття «педагогічні умови» (А. Литвин, Н. Тверезовська, Є. Хриков), варіативності наукових підходів до педагогічних умов формування методологічної, дослідницької компетентності фахівців різного профілю (Л. Бондаренко, М. Вінник, І. Козубцов, Н. Котенко, Н. Кугай, О. Лаврентьева, Лі Ліцюань, Л. Нічуговська, І. Олійник, М. Фролова та ін.) зумовили визначення та обґрунтування *педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання:*

- актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців;
- інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань;
- включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи;
- розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності.

Визначені педагогічні умови є взаємопов'язаними та взаємозалежними, відбивають сутнісні характеристики компонентів методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Перш ніж запропонувати та розробити педагогічні умови, які б ефективно впливали на процес формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання, вважаємо за необхідне спочатку провести аналіз освітньо-професійних програм (ОПП) та навчальних планів закладів вищої освіти України, які здійснюють професійну підготовку здобувачів першого (бакалаврського) рівня спеціальності «Комп'ютерна інженерія».

Для аналізу було обрано освітні програми таких закладів:

- Національний університет біоресурсів і природокористування України (ОПП «Комп'ютерна інженерія»);
- ДЗ «Луганський національний університет імені Тараса Шевченка» (ОПП «Комп'ютерна інженерія»);
- ДЗ «Київський авіаційний інститут» (ОПП «Комп'ютерні системи та мережі»);
- Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського» (ОПП «Системне програмування та спеціалізовані комп'ютерні системи», ОПП «Комп'ютерні системи та мережі»);
- Черкаський державний технологічний університет (ОПП «Комп'ютерні системи та мережі»);
- Державний університет інформаційно-комунікаційних технологій (ОПП «Комп'ютерна інженерія»);
- Харківський національний університет радіоелектроніки (ОПП «Комп'ютерна інженерія»).

Усі досліджені освітні програми відповідають вимогам чинного Стандарту вищої освіти України за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» першого (бакалаврського) рівня та мають

спільне концептуальне спрямування. За оновленими позначеннями галузей та спеціальностей – це спеціальність F 7 «Комп'ютерна інженерія», галузь знань F «Інформаційні технології». Важливо підкреслити, що кожна програма має власну унікальність. Наприклад, в Національному технічному університеті України «Київський політехнічний інститут імені Ігоря Сікорського» (КПІ ім. Ігоря Сікорського) акцент робиться на системному програмуванні та проєктуванні високонавантажених обчислювальних систем. У Харківському національному університеті радіоелектроніки (ХНУРЕ) підкреслюється здатність до аналізу та експлуатації програмно-апаратного забезпечення за допомогою сучасних методів моделювання. В Національному університеті біоресурсів і природокористування України (НУБіП України) ОП орієнтована на впровадження комп'ютерної інженерії з урахуванням викликів природокористування та агропромислової сфери, що вимагає від фахівця високої преадаптивності та гнучкості. У ДЗ «Луганський національний університет імені Тараса Шевченка» акцент зроблено на здатності проєктувати та підтримувати комп'ютерні системи та мережі різного призначення та використовувати сучасних віртуальних обчислювальних ресурсів.

Важливим аспектом аналізу освітніх програм професійної підготовки фахівців галузі 123 «Комп'ютерна інженерія» було виокремлення місця освітнього компоненту «Філософія». Доведено, що зазначений освітній компонент у більшості освітніх програм пропонується до вивчення на 2 або навіть 3 курсі, що зумовлює недостатній рівень формування в студентів загальнофілософських уявлень про особливості наукового та практико-зорієнтованого пізнання відповідної предметної галузі, що є критично важливим для осмислення загальноінженерних дисциплін. Відзначимо, що, на жаль, але порушується зв'язок між гуманітарною та фундаментальною підготовкою на початкових етапах навчання, що спричинює утруднення щодо адаптації майбутніх фахівців до етичних та філософських аспектів розвитку сучасних інформаційних технологій, штучного інтелекту й кібербезпеки, які мали б закладатися як світоглядний фундамент уже на 1 курсі. Зазначені позиції набувають особливої актуальності внаслідок того, що філософія формує базові

навички системного аналізу та логіки. Відповідно, якщо студент отримує ці інструменти лише на 2-3 курсі, то попередній рік-два він вивчає спеціальні дисципліни без належного світоглядного базису. Підкреслимо також, що лише окремі бакалаврські програми містять окремий освітній компонент, пов'язаний із методологією наукових досліджень (наприклад, в ДЗ «Луганський національний університет» для студентів 3 курсу відповідної спеціальності є ОК «Основи досліджень в ІТ-галузі»).

Зазначені позиції актуалізують необхідність додаткової уваги у процесі викладання фахових дисциплін до методологічних аспектів опанування предметною галуззю, до використання різних форм та методів формування системного та критичного мислення, усвідомлених позицій щодо загально-наукових та фахових методів вирішення конкретних професійних ситуацій з урахуванням можливостей технологій проблемно зорієнтованого навчання.

На основі аналізу ОПП нами було виокремлено пріоритетні напрями з метою їх порівняння у контексті підґрунтя для проблемно зорієнтованого навчання (Табл. 2.1).

Таблиця 2.1.

Пріоритетні напрями особливостей освітньо-професійних програм ЗВО

Особливості ОПП (спрямованість)	КП ім. Ігоря Сікорського	ЧДТУ	ДУКТ	ХНУРЕ	НУБіП України	КАІ	ЛНУ імені Тараса Шевченка
Вирішення комплексних інженерних задач в умовах невизначеності	+	+	+	+	+	+	+
Проблемно-зорієнтоване та проектне навчання	+	+	+	+	+	+	+
Системне програмування та низькорівневе проєктування	+	+-	-+	-	+	+-	+
Математичне та комп'ютерне моделювання систем	+	-+	+	+	+	-+	+

За результатами аналізу особливостей ОПП можемо стверджувати, що програми мають загальну орієнтацію на формування вмінь розв'язувати комплексні задачі. Окрім цього, важливим чинником є наявність загальних (ЗК) та фахових (ФК) компетентностей і програмних результатів навчання (ПРН). У всіх програмах чітко артикулюються:

- Здатність до абстрактного мислення, аналізу та синтезу (ЗК).
- Здатність проведення досліджень на відповідному рівні (ЗК).
- Здатність вирішувати проблеми у галузі комп'ютерних та інформаційних технологій, визначати обмеження цих технологій (ФК).
- Здатність аргументувати вибір методів розв'язування спеціалізованих задач та критично оцінювати отримані результати (ФК).

Всі зазначені програмні результати вимагають, перш за все, сформованої на високому рівні методологічної компетентності. Втім, незважаючи на наявність цих компетенцій у переліку, механізм їх формування не є наскрізним. Аналіз навчальних планів демонструє, що формування досліджуваної компетентності відбувається стихійно та несистематично в межах окремих фахових дисциплін. Наочно перелік таких дисциплін подано в Таблиці 2.2.

Таблиця 2.2.

Фахові дисципліни професійної підготовки, в межах яких може відбуватися фрагментарне формування методологічної компетентності

Дисципліни (напрямки)	КІП ім. Ігоря Сікорського	ЧДТУ	ДУКТ	ХНУРЕ	НУБіП України	КАІ	ЛНУ імені Тараса Шевченка
Системне програмування	+	+	+	+	+	+	+
Комп'ютерні мережі	+	+	+	+	+	+	+
Моделювання комп'ютерних систем / Системний аналіз	+	-	+	+	+	-	+
Теорія прийняття рішень та алгоритмізація	+	+	+	+	+	+	+
Проектний практикум (Курсове проектування)	+	+	+	+	+	+	+

Відповідно до узагальнювальної таблиці, можемо зробити такі висновки. В освітньо-професійних програмах всіх закладів вищої освіти доволі чітко представлено праксеологічно-технологічний компонент методологічної компетентності (через дисципліни програмування, апаратної інженерії, архітектури мереж). Більшість бакалаврських програм не містить обов'язкового освітнього компонента, який би безпосередньо та системно фокусувався на методології наукового чи інженерного пошуку (найчастіше такі дисципліни винесені на магістерський рівень). Відзначимо, що у процесі вивчення фундаментальних фахових дисциплін часткове усвідомлення методів дослідження відбувається, але дефіцит цілеспрямованої підготовки створює ризик того, що студенти опановують готові інструкції без глибокої рефлексії над природою інструментарію. Питання преадаптивності, самостійного концептуального конструювання та подолання технологічної невизначеності залишаються недостатньо висвітленими.

У межах аналізу освітньо-професійних програм ми дійшли висновку, що застосування технологій проблемно зорієнтованого навчання набуває особливої актуальності. Штучне та кероване моделювання реальних інженерних проблем під час практичних занять дозволить не просто передати студенту алгоритм дій, а активізує рефлексивну, когнітивну та прогностичну функції його методологічної компетентності. Це переконливо доводить своєчасність та необхідність розробки нашого наукового дослідження та впровадження відповідних педагогічних умов в освітній процес.

Перша педагогічна умова – актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців. Теоретичними засадами обґрунтування зазначеної умови визначено наукові положення щодо: сутності понять «мотив», «мотивація», психології мотивації особистості (С. Занюк, Т. Кудріна, А. Маслоу та ін.) [37; 66; 164; 199]; мотивації діяльності людини (З. Кісіль, Д. Швець) [45]; мотивації навчальної діяльності студентів ЗВО (Л. Кондратенко, І. Нікітіна, Н. Пророк та ін.) [99; 117]; мотивації навчально-пізнавальної діяльності студентів технічних спеціальностей

(В. Ключко, А. Коломієць) [46]; психологічних особливостей мотивації ІТ-спеціаліста (Д. Філоненко) [156].

У словникових виданнях мотив розглядається як «спонукання до діяльності, пов'язані із задоволенням потреб суб'єкта; сукупність зовнішніх або внутрішніх умов, що викликають активність суб'єкта і визначають її спрямованість», мотивація – як «усвідомлювані або неусвідомлювані психічні фактори, що спонукають індивіда до здійснення певних дій і визначають їхню спрямованість та цілі» [164, с. 264].

У фундаментальних працях А. Маслоу мотивація представлена як процес спонукання особистості до задоволення потреб, надав їй енергетичної функції та динамічного характеру [199]. Гуманістична теорія мотивації А. Маслоу характеризується цілісним поглядом на людину та її мотиви, динамічністю та ієрархічністю мотивів, що спрямовані на певну ціль, яка виникає з дефіциту або прагнення до росту

Т. Кудріна в контексті дослідження проблем психології мотивації підкреслює, що «життєвий сенс мотиву можна визначити його зв'язком з актуальними потребами, які реалізуються в діяльності, спрямованої на цей мотив» [66, с. 23]. Мотив «узгоджує задоволення потреби з теперішнім та майбутнім особистості, зі зберіганням її стійкості й сталості в умовах мінливого життя. Отже можна сказати, що мотив, крім спонукальних, має оцінні регуляторні й змістовні характеристики, які або зв'язані зі свідомістю, або містяться в сфері свідомості» [66, с. 47]. Важливим є зауваження С. Занюка про те, що мотивацію необхідно розглядати як сукупність чинників, які визначають активність особистості, а саме: мотиви, потреби, стимули, ситуативні чинники, які ініціюють, спрямовують та підтримують поведінку людини у конкретний момент. Дослідник підкреслює, що така сукупність викликає активність організму та визначає його націленість [37, с. 5].

Мотивація здобувачів вищої освіти до вибору професії є вагомим чинником успішності як навчальної, так й майбутньої професійної діяльності. Так, за даними опитування Асоціації ІТ Ukraine та EPAM University (2024)

(<https://happymonday.ua/ponad-polovyna-shkolyariv-hochut-pratsyuvaty-it>) 54% випускників школи бажають отримати професію в ІТ-галузі. Основними причинами такого вибору зазначено такі: стійкість галузі (школярі бачать, що попри повномасштабну війну, коли інші галузі постраждали сильніше, більшість айтівців продовжує працювати); прагнення до фінансової самостійності (частина молоді бажає почати працювати якомога раніше; в ІТ реальні знання та практичні навички цінуються більше, ніж наявність документа про освіту; школярі вже цікавляться можливістю працювати паралельно з навчанням). Д. Філоненко розглядає особливості мотивації ІТ-фахівців та підкреслює, що в них домінує потреба в досягненнях. Відповідно для забезпечення найвищої ефективності та задоволеності актуальними є: «внутрішня мотивація, в якій досягнення цілей виступає вищою нагородою, ніж фінансове; очікування можливостей кар'єрного росту. Створюючи продукт, вони очікують визнання результатів своєї праці важливими для них людьми з професійного та соціального середовища» [156, с. 246]. Важливою є й позиція автора, що «після досягнення успіху у розробці, підтримка продукту вже не викликає інтересу, отже для отримання задоволення від діяльності потрібні постійні інновації, додавання нових функцій в систему, що працює. Зворотний зв'язок від користувачів розроблених систем (як позитивний, так і негативний) має сильну мотивуючу дію на ІТ-фахівців, адже дозволяє їм переглянути свій прогрес і поліпшити систему. Тож, особиста відповідальність за вирішення проблем і несправностей у системі, також виступає мотиваційним чинником» [156, с. 246-247].

Зазначені особливості орієнтації ІТ-фахівців на внутрішню мотивацію, постійні інновації та особисту відповідальність за вирішення складних проблем безпосередньо зумовлює потребу у виході за межі суто технічних навичок (*hard skills*). Оскільки прагнення до досягнень і професійного визнання вимагає від спеціалістів здатності не просто створювати продукт, а системно оцінювати свій прогрес, гнучко реагувати на зворотний зв'язок від користувачів та вдосконалювати роботу всієї системи, виникає об'єктивна необхідність у формуванні їхньої методологічної компетентності. Саме ця компетентність стає

тим внутрішнім інструментом, який дозволяє фахівцям свідомо структурувати свою інноваційну діяльність, концептуалізувати отриманий досвід і самостійно знаходити ефективні алгоритми розв'язання нестандартних професійних завдань в умовах динамічних змін. Саме мотивація визначає суб'єктну позицію студента, перетворюючи процес навчання з пасивного накопичення інформації на активне опанування методології професійної діяльності.

Вагоме значення для розуміння сутнісних характеристик мотивації здобувачів вищої освіти має урахування значущості внутрішніх та зовнішніх мотивів навчальної та професійної діяльності. Внутрішня мотивація є найбільш стійким чинником і виявляється у формі пізнавального інтересу до складності алгоритмічних задач та задоволення від процесу інтелектуального пошуку. Цей вид мотивації характеризується відчуттям автономії та власної компетентності, коли студент усвідомлює, що володіння методологією дає йому владу над технологією, а не навпаки. Особливе місце посідає потреба в професійній суб'єктності, коли студент прагне оволодіти методологічними засадами (наприклад, системним аналізом чи теорією алгоритмів), щоб звільнитися від залежності від конкретних програмних продуктів, які швидко застарівають.

Зовнішня мотивація у сфері комп'ютерної інженерії має потужний прагматичний контекст, який зумовлений високою динамікою ІТ-ринку, де рівень заробітної плати та престижність посади безпосередньо корелюють із глибиною експертизи фахівця. Розуміння того, що методологічні знання та вміння є гарантом конкурентоспроможності та кар'єрного успіху, стимулює студентів до опанування складних теоретичних концептів. Однак для сталого розвитку важливо, щоб зовнішні стимули (диплом, висока зарплата) поступово трансформувалися у професійно-ціннісні орієнтації – усвідомлення соціальної значущості інженерної праці, етики розробника, відповідальності за надійність створених систем.

На нашу думку, актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок як засобу професійного становлення та кар'єрного зростання в майбутніх фахівців з комп'ютерної інженерії пов'язано із

ознайомленням на початковому етапі навчання із вимогами професійного стандарту, переліком компетенцій у посадових інструкціях, ресурсами агенцій з добору персоналу у контексті перспектив працевлаштування та можливостей кар'єрного розвитку. Важливими є проведення мотиваційних тренінгів, презентація досягнень відомих фахівців ІТ-галузі (вітчизняних та зарубіжних) як підґрунтя для рефлексивної діяльності та визначення власних шляхів самореалізації в професії.

Практичні шляхи для формування мотивації вимагають відходу від традиційних лекційних форматів на користь методів, що імітують реальну професійну діяльність. Основним механізмом тут виступає проблематизація комп'ютерної інженерної дійсності, використання кейс-технології, проєктне навчання, яке дозволяє студенту побачити методологію не як абстрактну теорію, а як інструмент розв'язання конкретної інженерної проблеми. Також актуальним технологічним аспектом є гейміфікація та змагальність, що можуть бути реалізовані через участь у хакатонах, олімпіадах з програмування та командних челенджах. Такі формати створюють ситуацію успіху та здорової конкуренції, де володіння досконалою методологією (наприклад, знання патернів проєктування або методів оптимізації коду) стає вирішальною перевагою. Це стимулює студентів до самостійного поглибленого вивчення матеріалу поза межами обов'язкової програми. Особливе значення має впровадження кейс-технології, що передбачає аналіз реальних технічних аварій або успішних стартапів у сфері комп'ютерної інженерії. Розбір причин збоїв у складних системах через недоліки в методології їх проєктування формує в майбутніх фахівців професійну відповідальність та усвідомлення критичної важливості методологічної грамотності.

Стратегічним фундаментом для трансформації пасивного засвоєння знань в активний процес методологічного пошуку доцільно розглядати проблемно зорієнтоване навчання. У контексті підготовки фахівців із комп'ютерної інженерії важливість цього підходу зумовлена специфікою самої галузі, де *розв'язання будь-якої інженерної задачі починається не з написання коду, а з аналізу суперечностей, декомпозиції проблем та вибору адекватних моделей*

дослідження. Вирішуючи реальні кейси, запозичені з практики провідних технологічних компаній, студент відчуває себе причетним до фахової спільноти. Це підкріплює соціальну мотивацію та прагнення до самореалізації, оскільки успішне розв'язання складної проблеми дає відчуття власної компетентності та інтелектуальної спроможності.

Друга педагогічна умова – інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань. Теоретичними засадами її обґрунтування визначено наукові положення щодо: сутнісних ознак методологічного знання, методологічної підготовки здобувачів вищої освіти в ЗВО (Л. Бутенко, Л. Ваховський, А. Вихрущ, В. Вихрущ, С. Гончаренко, Л. Ковальчук, Н. Кугай, О. Лаврентьєва, С. Сисоєва, Я. Топольник та ін.) [11; 13; 15; 16; 26; 27; 50; 70; 131; 150; 152]; викладання філософії у технічних ЗВО (М. Колотило, О. Кузнецов [55; 67]; теоретико-методичних засад професійної підготовки майбутніх фахівців з комп'ютерної інженерії (М. Бирка, М. Вінник, М.-О. Єршов, В. Круглик, С. Литвинова, Н. Морзе, С. Проскура, М. Семенов, С. Семеріков, І. Синельник, Я. Сікора, К. Стрюк та ін.) [4; 17; 34; 62; 96; 115; 127; 128; 130; 136; 145]; теорії та практики розробки та використання проблемних завдань у вищій школі (Г. Барроуз, Т. Баррет, С. Хмело-Сільвер та ін. [169; 170-172; 192; 217].

Передумовою успішної методологічної підготовки майбутніх фахівців з комп'ютерної інженерії вважаємо взаємозв'язок загальнокультурних та фахових складників професійного становлення здобувачів вищої освіти, зокрема актуалізацію філософських знань в контексті специфіки відповідної предметної галузі. Вище було зазначено, що освітній компонент «Філософія» не повною мірою реалізує свої можливості у методологічній підготовці майбутніх фахівців, хоча, як підкреслює О. Кузнецов, «за допомогою філософії формуються навички розумного міркування – самостійного, доказового, контекстуального, критичного – для того, щоб підготувати інтелект до засвоєння нового знання і виховати морально і соціально відповідальну особистість» [67, с. 84]. У фундаментальній збірці відомого українського філософа І. Добронравової особливо наголошується

на переході науки до постнекласичного етапу, де об'єктами дослідження та проєктування стають складні системи, що розвиваються та включають саму людину [31]. Концепція практичної філософії науки І. Добронравової актуалізує необхідність філософської підготовки майбутніх фахівців, оскільки сучасна ІТ-інфраструктура функціонує за законами самоорганізації, демонструючи точки біфуркації, емерджентність та чутливість до мінімальних збоїв. Оскільки цифрова реальність безпосередньо впливає на суспільні процеси, кібербезпеку, конфіденційність, то майбутній фахівець не може залишатися вузьким технократом, а повинен чітко усвідомлювати соціальну й етичну відповідальність за свої розробки та розуміти межі застосовності алгоритмів.

За умови відсутності на початковому етапі навчання обов'язкового курсу «Філософія» суттєве навантаження щодо розкриття значущості стратегій пізнання, загальнонаукових методів у професійній та дослідницькій діяльності майбутніх фахівців покладається на базові дисципліни професійного блоку. Так, у межах ОПП «Комп'ютерна інженерія» фахова підготовка розпочинається із курсу *«Вища математика»*, який створює первинну понятійну базу, яка дає змогу переводити абстрактні фізичні чи інформаційні процеси у чіткі формалізовані моделі. Саме на етапі опанування вищої математики здобувач вищої освіти засвоює головний інженерний принцип: будь-яку складну систему можна декомпонувати, описати мовою строгих закономірностей і оптимізувати на основі фундаментальних законів. Необхідно підкреслити, що аналітичний апарат вищої математики формує системне та структурно-логічне мислення. Вивчення систем лінійних рівнянь, векторної алгебри та теорії матриць безпосередньо закладає розуміння багатьох концепцій комп'ютерної архітектури, починаючи від паралельних обчислень і графічних процесорів до обробки багатовимірних масивів даних. Суттєвим аспектом методологічної підготовки є залучення теорії ймовірностей та математичної статистики.

Важливим аспектом початкового етапу навчання майбутніх фахівців з комп'ютерної інженерії є актуалізація знань щодо загальнонаукових методів пізнання (аналіз, синтез, індукція, дедукція, узагальнення, класифікація тощо). У

словнику В. Шапара актуалізація (від лат. *actualis* – діяльний) розглядається, як «дія, що полягає у викликанні засвоєного матеріалу з довгострокової або короткочасної пам'яті для наступного використання його при впізнаванні, пригадуванні, спогаді або при безпосередньому відтворенні» [164, с. 13].

Здобувачі вищої освіти вже мають певне уявлення про загальнонаукові методи пізнання за результатами шкільного навчання. Відповідно, актуалізація знань про такі методи здійснюється різними шляхами. По-перше, демонстрація використання загальнонаукових методів у розв'язанні прикладної задачі. По-друге, наскрізне використання методів моделювання (наприклад, побудова алгоритмічних моделей для опису процесів обробки даних, що демонструє метод аналогії та формалізації на практиці). По-третє, використання завдань у формі відкритих інженерних проблем, вирішення яких вимагає застосування гіпотетико-дедуктивного методу, формулювання гіпотези та дедуктивних висновків (Чому система працює некоректно чи дає збій? Які наслідки має відповідна гіпотеза?), експериментальної перевірки (покрокове локалізування проблеми за допомогою системного моніторингу та тестування).

Таким чином, практична реалізація інтеграції фундаментальних знань із профільними дисциплінами комп'ютерної інженерії спирається на впровадження активних форм і методів навчання, що трансформують теорію у прикладні інструменти – виконання міждисциплінарних проєктів, завдання щодо проблематизації інженерної дійсності, кейс-метод на основі комплексу кейсів різного типу.

Міждисциплінарна інтеграція на основі технологій проблемно зорієнтованого навчання виступає важливою передумовою успішності реалізації завдань дослідження, оскільки сприяє подоланню ключової суперечності вищої технічної школи між цілісним, багатокритеріальним характером реальної інженерної діяльності та традиційною предметно-центричною, фрагментованою моделлю викладання. Якщо фундаментальні принципи наукового пізнання та системного аналізу вивчаються ізольовано від прикладного інструментарію комп'ютерної інженерії, вони неминуче сприймаються здобувачами як абстрактна філософська надбудова, позбавлена практичної цінності. Надзвичайна

важливість такої інтеграції переконливо доводиться крізь призму концепції фундаменталізації освіти. Видатний український педагог С. Гончаренко відзначав, що фундаменталізація освіти означає її спрямування на універсальні знання, а її завданням є створення оптимального середовища для виховання гнучкого багатогранного мислення і сучасної методології осмислення дійсності [26, с. 88]. С. Семеріков обґрунтовує такі напрями та функції фундаменталізації навчання інформатичних дисциплін у вищій школі: «опанування методологічно важливими та інваріантними знаннями, що мають довгий термін життя, необхідними для професійної діяльності фахівця в галузі інформаційних технологій (*методологічна функція*); тісний зв'язок інформатичної освіти з професійною практичною діяльністю (*професійно-орієнтувальна функція*); розвиток творчої і пізнавальної активності та самостійності (*розвивальна функція*); розвиток методичних систем навчання інформатичних дисциплін з врахуванням перспектив розвитку «економіки знань» та інформаційного суспільства (*прогностична функція*); системність засвоєння інформатичних дисциплін на основі глибокого розуміння сучасних проблем інформатики (*інтегративна функція*)» [128, с. 67].

У реаліях комп'ютерної інженерії це означає, що студент має збагнути приховані взаємозв'язки між математичною логікою, апаратною архітектурою та високорівневим програмним забезпеченням. Дослідники наголошують на необхідності формування цілісного, системного світогляду майбутнього інженера, реалізація якого неможлива в межах замкнених інформаційних кластерів окремих кафедр. Організаційним каталізатором та середовищем такої інтеграції у нашому дослідженні виступають *міждисциплінарні проблемні завдання*. Так М. Брасслер (M. Brassler) та Я. Деттмерс (J. Dettmers) [177] доводять, що саме міждисциплінарне проблемно зорієнтоване навчання (iPBL) забезпечує значно вищий рівень розвитку компетентностей, ніж традиційне проєктне навчання (iPjBL). Це зумовлено фокусом на самому процесі пізнання, а не на створенні кінцевого матеріального продукту. У контексті комп'ютерної інженерії це означає, що замість механічного виконання технічного проєкту, проблемне

завдання змушує їх ітеративно проходити через чіткі методологічні кроки: аналіз понятійного апарату фундаментальних та прикладних дисциплін; формулювання інтегрованих цілей навчання, самостійне дослідження наукових джерел на стику напрямків, створення спільного синтезованого командного висновку.

У аналітичному огляд А. Ван ден Беємта (A. Van den Beemt) [219] йдеться про різні моделі використання міждисциплінарності. Т-подібна модель навчання (T-shape) – фундаментальні міждисциплінарні концепції та системне мислення (горизонтальна вісь «Т») поєднуються з глибокими технічними знаннями безпосередньо з інженерії (вертикальна вісь «Т»). Модель базована на виконанні комплексного інженерного проєкту (*Capstone design course*), передбачає залучення до командної роботи.

Відзначимо наукові положення М. Уерта-Гомес-Меродіо (M. Huerta-Gomez-Merodio) та М.-В. Рекена-Гарсія-Крус (M.-V. Requena-Garcia-Cruz) [193], які доводять, що традиційна структурна фрагментація навчальних планів вищої школи призводить до виникнення явища «інертних знань», за якого здобувачі спроможні відтворювати теоретичні концепції, але виявляються безпорадними під час їхнього перенесення у практичну площину. Запропонована дослідниками наскрізна, практико зорієнтована методологія успішно інтегрує теоретичний зміст із професійними компетентностями на засадах конструктивізму, проблемно зорієнтованого навчання, концепції емпіричного навчання Д. Колба та міжнародних інженерних стандартів CDIO.

У контексті підготовки майбутніх фахівців з комп'ютерної інженерії екстраполяція такого підходу вимагає розробки специфічних *міждисциплінарних сценаріїв, які б органічно поєднували апаратну, програмну та мережеву складові*. Яскравим прикладом такого інтегрованого проблемного завдання є проєктування захищеної системи збору телеметрії для пристроїв Інтернету речей на базі мікроконтролерів (наприклад, архітектури ESP32). Вирішення цієї проблеми неможливе в межах однієї дисципліни, оскільки здобувач освіти змушений одночасно застосовувати знання з комп'ютерної електроніки для розуміння апаратних обмежень датчиків, залучати методи системного програму-

вання для оптимізації обробки сигналів, впроваджувати криптографічні протоколи для забезпечення безпечної передачі даних через інтернет-канали.

Не менш дієвим є конструювання проблемних ситуацій у сфері мережевої безпеки та адміністрування. Розв'язання задачі із забезпечення стійкості складної корпоративної мережі, змодельованої за допомогою індустріальних інструментів рівня Cisco Packet Tracer, або завдання із захисту агентів федеративного машинного навчання від цілеспрямованих кібератак типу «отруєння даних», вимагає цілісного підходу. Студенти повинні об'єднати принципи побудови топологій маршрутизації, алгоритми обробки масивів даних та актуальні стандарти кіберзахисту. Оволодіння цими інтегрованими знаннями в процесі розв'язання відкритої проблеми формує у майбутніх інженерів здатність бачити обчислювальну екосистему як єдиний організм, де зміна одного апаратного чи програмного параметра неминує впливає на життєздатність усієї архітектури.

Третя педагогічна умова – включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи. Теоретичними засадами обґрунтування зазначеної умови визначено наукові положення щодо: діяльнісного підходу в освіті (І. Бех, О. Дубасенюк, Л. Кушнір, Л. Шелестова, М. Прінс (M. Prince) та ін.) [2; 30; 68; 165; 203], праксеологічного підходу в освіті (І. Мельничук, Л. Романовська, Ю. Тулашвілі, Т. Котарбінський (T. Kotarbinski) та ін.) [92; 119; 153; 197], проблемно зорієнтованого навчання у вищій школі (Н. Дмітренко, Л. Козуля, Ю. Колесник, Г. Луценко, О. Романовський та ін.) [54; 79; 114; 120], проблемно зорієнтованого навчання у підготовці фахівців ІТ-сфери (О. Баранюк) [1], технологій проблематизації в освіті (М. Фуко (M. Foucault); Н. Бондаренко, Л. Бутенко, С. Косянчук та ін.) [186; 9; 10; 11], кейс-технології як складника проблемно зорієнтованого навчання (С. Постова, Б. Пустовойт, Т. Ситнік, Ю. Сурмін, В. Федів; Г. Барроуз; Елейн Ю (E. Yew), К. Го (K. Goh) та ін.) [112; 118; 134; 146; 155; 171; 222].

Теоретичне обґрунтування зазначеної педагогічної умови передбачає логічний взаємозв'язок діяльнісного та праксеологічного підходів. Діяльнісний підхід у цьому контексті виступає фундаментальною основою, оскільки трактує процес навчання як цілеспрямовану, активну систему взаємодії, де формування методологічних умінь, навичок командної роботи та професійного мислення відбувається безпосередньо в процесі трансформації теоретичних знань у практичні дії. Відповідно, праксеологічний підхід зміщує фокус на категорії ефективності, раціональності та результативності людської праці в умовах реальних викликів ІТ-індустрії. Взаємодоповнення цих методологічних орієнтирів дає змогу розглядати квазіпрофесійне середовище не просто як імітацію виробничих процесів, а як праксеологічно вивірений простір, де моделювання критичних ситуацій і командна розробка програмно-апаратних рішень забезпечують максимальну готовність майбутніх інженерів до успішної, свідомої та високопродуктивної професійної діяльності.

У підрозділі 1.4. представлено концептуальні засади проблемно зорієнтованого навчання у вітчизняному та зарубіжному науковому дискурсі. Відзначимо ще раз, що ключовими концептами у проблемно зорієнтованому навчанні є поняття «проблема», «проблемна ситуація», «проблематизація». Сутнісні ознаки таких понять представлено у словникових виданнях. Так, поняття «*проблема*» потлумачується так: «в широкому розумінні складне теоретичне або практичне або практичне питання, що потребує розв'язання, вивчення й дослідження; у науці – суперечлива ситуація, що виступає у вигляді протилежних позицій у поясненні будь-яких явищ, об'єктів, процесів і потребує адекватної теорії для її розв'язання» [25, с. 271]; усвідомлення неможливості вирішити труднощі й протиріччя, що виникли вданій ситуації, засобами наявного знання і досвіду» [164, с. 371]. *Проблемна ситуація* розглядається як: «ситуація, для оволодіння якою окремий суб'єкт (або колектив) має знайти й застосувати нові для себе знання чи способи дій» [25, с. 271]; психологічна категорія, зумовлює «лише початкову стадію розумової взаємодії суб'єкта з об'єктом – стадію, пов'язану з породженням пізнавального мотиву і висуванням попередніх

гіпотез про способи вирішення проблемної ситуації. Перевірка цих гіпотез призводить до того, що проблемна ситуація перетворюється на проблему або задачу [164, с. 371].

Сучасні наукові розвідки щодо *поняття «проблематизація»* ураховують класичні позиції М. Фуко, що проблематизація – це не відкриття того, що вже існувало, і не вигадкування проблеми на порожньому місці, це «ансамбль дискурсивних і недискурсивних практик», через які об'єкт вводиться в гру істини та хибності, стаючи об'єктом думки й аналізу [186]. У такому контексті відзначимо дуже яскраву назву статті Бу Акар Хіби [191] – *«Якщо ви не проблематизуєте це, ви не побачите і не зрозумієте»*. Відповідно, у контексті вирішення завдань професійної підготовки здобувачів вищої освіти проблематизація розглядається, як «визначення, фіксація базових викликів (перешкод, точок розриву, бар'єрів), подолання яких сприятиме розвитку особистості та професійної діяльності людини в цілому» [10, с. 23], елемент проблематизованого навчання, у межах якого реалізується «дослідницька діяльність учня, стимульована певною ситуацією, що спонукає його ставити питання-проблеми, формулювати гіпотези й перевіряти їх під час розумових і практичних дій» [9, с. 163]. Відзначимо також положення про те, що проблематизація пов'язана із процесом «ідентифікації, ставлення під сумнів і критичного аналізу визначальних аспектів поняття, практики чи ситуації, які часто сприймаються як належне» [191].

У контексті професійної підготовки майбутніх фахівців з комп'ютерної інженерії проблематизація є важливим аспектом, тому що сприяє трансформації процесу навчання від пасивного засвоєння готових знань до активного пошуку інноваційних рішень, розвитку критичного та інженерного мислення, адаптивності до технологічних змін, включенню у симуляцію реального виробничого середовища, формуванню мотивації до дослідницької діяльності. Отже, проблематизація забезпечує зміщенню фокусу із питання *«Що я маю знати?»* на питання *«Як я можу розв'язати цю проблему?»*, що й визначає кваліфікаційну зрілість сучасного інженера.

Дослідники виокремлюють різні технології проблематизації: перехід в термінологічне поле іншої науки, метод емпатії «Подивіться на світ чужими очима», метод шести капелюхів Едварда де Боно, створення метафоричного образу проблеми, ресурсний розворот (погляд на обставини і проблеми з іншої точки зору), рефреймінг проблем (переформулювання), «диверсійний аналіз» [10].

Типологія завдань щодо проблематизації комп'ютерної інженерної дійсності будується навколо виявлення та структурування прихованих суперечностей, технологічних обмежень і дефіцитів у тривалій взаємодії «апаратне забезпечення – програмний код – користувач/середовище». Оскільки інженерна дійсність є комплексною, завдання з проблематизації доцільно класифікувати за об'єктом аналізу, типом системного конфлікту та рівнем абстракції. З урахуванням специфіки професійної діяльності майбутніх фахівців з комп'ютерної інженерії доцільно виокремити такі завдання:

- *архітектурно-структурні*, спрямовані на виявлення та критичний аналіз фундаментальних системних суперечностей між масштабованістю, рівнем функціональної зв'язності та загальною надійністю проєктованих об'єктів;
- *ресурсно-технологічні*, сутність яких пов'язана із проблематизацією жорстких фізичних та алгоритмічних лімітів сучасної обчислювальної інфраструктури;
- *завдання інтеграційної та еволюційної сумісності* спрямовані на подолання когнітивного спрощення, коли здобувач очікує ідеальних умов розробки «з нуля». Майбутній фахівець занурюється у суперечливу інженерну дійсність, де виникає гострий конфлікт між прагненням впровадити інноваційні рішення та вимогою забезпечення зворотної сумісності з інфраструктурою, яка вже функціонує;
- *завдання інженерної безпеки та стійкості* орієнтовані на ситуації, коли фахова дійсність проблематизується через призму потенційних загроз і сценаріїв аномальної поведінки обчислювальних систем. Відповідно, майбутній інженер вчиться сприймати апаратне та програмне забезпечення не як ізольований ідеальний механізм, а як об'єкт постійного деструктивного

впливу, де будь-яка неврахована деталь або проєктний компроміс неодмінно стає критичною точкою компрометації;

- *людино-машинні та контекстуальні завдання* спрямовані на експлікацію та розв’язання суперечностей, що виникають на межі взаємодії технічної системи з її кінцевим користувачем, оператором або бізнес-середовищем.

Важливим складником вирішення завдань дослідження є **використання кейс-технології**, у межах якої надаються структуровані описи реальних технічних, організаційних ситуацій, що містять приховане чи явне протиріччя, дефіцит ресурсів або системний збій. Це дозволяє студенту зануритися у професійний контекст, де розв’язання інженерної проблеми вимагає не просто відтворення алгоритму, а багатокритеріального аналізу, синтезу альтернативних варіантів та прийняття відповідальних рішень в умовах невизначеності.

У контексті проблематики кейс-технологій використовуються поняття: *метод ситуаційного навчання* як «навчальний метод, застосування якого передбачає осмислення студентами реальної життєвої ситуації. Опис цієї ситуації одночасно відображає не тільки певну практичну проблему, але й актуалізує визначений комплекс знань, який необхідно засвоїти для її вирішення. При цьому сама проблема не має однозначних рішень, являє собою своєрідний розгалужений і неоднозначний оптимум» [135, с. 9]; *кейс-метод, або метод кейс-стаді* – «детальне дослідження окремого випадку, що належить до деякого класу феноменів» [146, с. 19]. До *технологій аналізу ситуацій* Н. Волкова відносить такі: метод традиційного аналізу конкретних ситуацій (ситуаційні завдання, ситуаційні вправи); метод ситуаційного навчання – кейс-стаді, метод кейсів, метод інциденту; метод розігрування ролей [18, с. 62-77].

В англomовній науковій літературі послуговуються, по-перше, поняттям «*Case-Study*», що розглядається як активна форма навчання, що зосереджена на випадку та передбачає навчання студентів на практиці; випадки – це реальні або вигадані історії, які розповідають про події, проблеми, дилеми, теоретичні чи концептуальні проблеми, які потребують аналізу та/або прийняття рішень [181]; по-друге, поняттям «*Case-Based Learning*», яке потлумачується як проблемно

зорієнтоване та контекстуалізоване навчання, що дозволяє студентам встановити зв'язки між теоріями та практикою. Вивчаючи та аналізуючи реальні проблеми/ кейси, студенти усвідомлюють, як абстрактні поняття використовуються або проявляються в реальних ситуаціях [182, с. 655]. Відомий американський дослідник К. Геррейд (C. Herreid), директор Національного центру кейс-методів у навчанні природничих наук (*National Center for Case Study Teaching in Science*) зазначає, що ефективний навчальний кейс має ґрунтуватися на актуальній проблемі, починатися із захопливої історії, містити конфлікт чи дилему, що не мають одного очевидного чи правильного розв'язання, адже саме це стимулює глибоку групову дискусію, спонукає аудиторію до активного аналізу та пошуку рішень [190].

Ю. Сурмін відзначає, що як інтерактивний метод навчання «Case study» «сприяє розвитку вміння вирішувати проблеми з урахуванням конкретних умов і за наявності фактичної інформації», «розвиває такі кваліфікаційні характеристики особистості, як здатність до аналізу і діагностики проблем, вміння чітко формулювати і висловлювати свою позицію, спілкуватися, дискутувати, сприймати й оцінювати інформацію, яка надходить у вербальній і невербальній формах [146, с. 20]. Важливим функціями кейс-методу є «навчальна, виховна, тренувальна, аналітична, дослідницька, систематизуюча, прогностична» [146, с. 21]. У науковій розвідці Т. Ситнік наголошується, що «кейс-метод виступає дієвим інструментом, який уможливорює використовувати набуті теоретичні знання для розв'язання практичних завдань, розвиток самостійного мислення у студентів, аргументовано висловлювати свою позицію. За допомогою цього методу студенти мають змогу проявити та вдосконалити аналітичні вміння й навички, знаходити найбільш раціональні рішення ситуації» [134, с. 71]. Використання кейс-методу у цифровому освітньому середовищі розглядають С. Постова, Н. Опанасюк та А. Опанасюк [112].

Актуальність використання кейс-технології/кейс-методу у процесі професійної підготовки майбутніх фахівців ІТ-сфери відзначається такими дослідниками, як Я. Сікора [136, с. 320], Л. Слободянюк [138, с. 88], К. Стрюк

[145, с. 114-115], В. Круглик [62, с. 45], Л. Макаренко та Ю. Шпильовий [80, с. 82], О. Нужин [102, с. 93] та ін.

Розробка та використання кейсів потребує цілеспрямованої роботи з урахуванням специфіки предметної галузі, рівня сформованості аналітичних умінь та навичок студентів, досвіду проблематизації. Процес опрацювання кейса охоплює етапи індивідуальної аналітики, командної дискусії, що імітує професійну взаємодію в інженерному колективі, та публічного захисту розробленої технічної пропозиції перед умовним замовником. Специфічна позиція кейс-технології в ІТ-освіті полягає також в її здатності інтегрувати програмну симуляцію з аналітичною роботою, коли студенти верифікують свої гіпотези через спеціалізоване ПЗ для проєктування схем або моделювання мережових протоколів. Такий підхід забезпечує формування не лише фахових компетентностей, а й критично важливих м'яких навичок (*soft skills*), таких як здатність аргументувати вибір технологічного рішення, оцінювати ризики кібербезпеки та враховувати економічну доцільність інженерних інновацій. Таким чином, кейс-метод стає ефективним механізмом професійної соціалізації, що готує майбутнього спеціаліста до вирішення реальних викликів високотехнологічного середовища.

Типологія кейсів у системі підготовки фахівців з комп'ютерної інженерії вибудовується відповідно до структури професійної діяльності інженера, що охоплює проєктування, експлуатацію, діагностику та модернізацію обчислювальних систем. Кожен тип кейса спрямований на формування специфічного кластера компетентностей, що корелюють із реальними викликами ІТ-індустрії.

Науково-дослідні та аналітичні кейси спрямовані на розвиток здатності до глибокого аналізу архітектурних рішень та вибору оптимального технологічного стеку. Студенти не створюють продукт власноруч, а виступають у ролі експертів, які оцінюють існуючі системи або концепції. Наприклад, аналіз порівняльної ефективності різних типів пам'яті (DRAM vs SRAM) для специфічних вбудованих систем або дослідження вразливостей у протоколах передачі даних. Відповідно, основна мета – навчити майбутнього інженера працювати з технічною документацією та науковими даними для прийняття обґрунтованих рішень.

Техніко-діагностичні кейси імітують ситуації системних збоїв або критичних помилок у роботі апаратного чи програмного забезпечення. Студенту надається опис стану системи в момент відмови (логи сервера, дампи пам'яті, показники датчиків температури чи напруги). Завдання полягає у виявленні першопричини несправності та розробці алгоритму її усунення. Це критично важливо для дисциплін, пов'язаних із обслуговуванням комп'ютерних мереж та адмініструванням систем.

Проектно-конструкторські кейси вимагають від студента синтезу знань для створення нового інженерного продукту. Кейс зазвичай містить технічне завдання від умовного замовника з жорсткими обмеженнями: бюджет, габарити, енергоспоживання, сумісність із застарілим обладнанням тощо. Вирішення такого кейса часто завершується побудовою логічної схеми або написанням високорівневої специфікації.

Ситуаційно-стратегічні кейси фокусуються на управлінських та економічних аспектах інженерної діяльності, розглядають комп'ютерну інженерію не як ізольований процес, а як частину бізнес-середовища. Сюди відносяться завдання з масштабування інфраструктури підприємства, вибору між закупівлею власного серверного обладнання та орендою хмарних потужностей, або оцінки ризиків при переході на нову операційну систему в масштабах корпорації. Такі справи формують стратегічне мислення та розуміння вартості володіння технологіями.

Кейси «Ретроспективного аналізу» засновані на реальних історичних інцидентах глобального масштабу. Аналіз причин аварії космічних апаратів через помилки в ПЗ, масштабних витоків даних або глобальних збоїв у роботі DNS-серверів дозволяє студентам усвідомити ціну інженерної помилки. Ретроспективний аналіз навчає передбачати каскадні ефекти в складних системах, розробляти стратегії відмовобезпечності.

Оцінювання результатів розв'язання кейсів вимагає відходу від бінарної системи «правильно / неправильно». Оскільки інженерна діяльність за своєю суттю є багатокритеріальною оптимізацією, успішність студента має вимірю-

ватися через сукупність технічних, економічних та системних показників. Такими критеріями оцінювання доцільно визначити такі: технічна валідність й обґрунтованість рішення (відповідність обраних апаратних засобів поставленому завданню, дотримання галузевих протоколів та стандартів безпеки, відсутність «вузьких місць», які можуть призвести до відмови системи за умов, описаних у кейсі), оптимальність та ефективність використання ресурсів (ресурсна економічність, енергоефективність, економічна доцільність), якість діагностичного аналізу (логіка пошуку, інтерпретація даних, запобігання рецидивам), якість оформлення технічної документації. Важливо також враховувати системність та продуктивність мислення, здатність студента бачити систему в динаміці та у взаємодії з іншими компонентами.

Таким чином, використання кейс-технології як інструменту проблемно зорієнтованого навчання у підготовці майбутніх фахівців з комп'ютерної інженерії дозволяє трансформувати освітній процес із репродуктивного засвоєння теорії на активне формування практичних компетенцій. На відміну від класичних академічних завдань, робота з реальними технічними сценаріями стає фундаментом для розвитку системного мислення, оскільки вчить студентів аналізувати архітектуру комп'ютерних систем через призму критичних ситуацій. Такий підхід ефективно поєднує розвиток вузькотехнічних навичок із формуванням важливих м'яких компетенцій, імітуючи умови реального конструкторського бюро або ІТ-департаменту, де успіх проєкту залежить від командної комунікації та здатності аргументувати складні технічні рішення. Ключова цінність кейс-методу в інженерній освіті полягає у зміщенні акценту з пошуку однієї правильної відповіді на процес глибинної діагностики та обґрунтованого вибору стратегії, що суттєво впливає на підготовку майбутніх фахівців до професійної діяльності в умовах невизначеності та стрімких технологічних змін.

Квазіпрофесійна діяльність у процесі формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії передбачає використання форм та методів роботи, спрямованих як на опанування методами інженерної діяльності, так й на формування комунікативних умінь та навичок командної

роботи як важливих складників soft skills сучасних IT-фахівців. Відзначимо доцільність організації хакатонів (англ. *hackathon*, від *hack* – хакер та *marathon* – марафон) – технологічних заходів, де програмісти, дизайнери, проєктні менеджери та інші IT-фахівці об'єднуються для створення новаторських проєктів за короткий період часу, воркшопів – інтенсивних практичних семінарів, де учасники здобувають навички через реальну дію, симуляційних практикумів, зокрема із використанням інструментів штучного інтелекту.

У книзі Дж. Сонмеза (J. Sonmez), яка на сьогодні є світовим бестселером, «*Soft Skills: The software developer's life manual*» [212] особливу увагу акцентовано саме на комунікативних навичках IT-фахівців, уміннях працювати в команді. Автор наголошує, що «факт полягає в тому, що більшу частину нашого часу у сфері розробки програмного забезпечення ми витрачаємо на спілкування з людьми, а не з комп'ютерами. Навіть код, який ми пишемо, створюється насамперед для сприйняття людьми, і лише в другу чергу – для розуміння комп'ютером...Якщо ви хочете бути хорошим розробником програмного забезпечення, ви повинні навчитися ефективно взаємодіяти з людьми (навіть якщо написання коду – це та частина вашої роботи, від якої ви отримуєте найбільше задоволення)» [212, с. 18].

Залучення до різних форм командної роботи сприяє усвідомленню сутності команди як групи людей, «що взаємодоповнюють один одного в ході досягнення загальних результатів, і які використовують особливу форму організації спільної діяльності. Команда заснована на продуктивній взаємодії учасників, які мають узгоджене бачення ситуації та стратегічних цілей команди та володіють відпрацьованими процедурами взаємодії» [20, с. 43].

Необхідність навичок командної роботи засвідчують вітчизняні дослідники питань професійної підготовки фахівців IT-сфери – К. Стрюк [145, с. 121], Л. Слободянюк [138, с. 47] та ін.

Командна робота здобувачів вищої освіти під час участі у хакатонах, воркшопах, аналізі кейсів різного типу сприяє запитувальній активності, формуванню аналітичних здібностей, критичного мислення як важливих

показників методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Четверта педагогічна умова – розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності. Теоретичними засадами обґрунтування зазначеної умови визначено наукові положення щодо: рефлексії як умови розвитку особистості (Л. Гапоненко) [21]; професійної рефлексії фахівців (О. Лопатюк, Т. Плачинда та ін.) [76; 110], методу потрфолію (Е. Ясинська, І. Навчук, Н. Литвинюк, М. Миронюк, Я.Сікора; І. Сімерманова (I. Cimermanova) та Д. Єславич (D. Jelavic) та ін.) [136; 168; 179],

Методологічна компетентність не є пасивним багажем знань, а формується лише тоді, коли студент стає активним суб'єктом власного навчання. Стимулювання рефлексивної позиції перетворює зовнішній зміст освіти (теорії, методи, підходи) на внутрішні професійні переконання. Ми розглядаємо процес формування методологічної компетентності майбутнього фахівців з комп'ютерної інженерії як динамічний етапний процес, який передбачає включення здобувачів до ретроспективної рефлексії на початковому етапі навчання (аналіз наявного досвіду, стереотипів щодо ІТ-сфери), ситуативної рефлексії на діяльнісному етапі (самоконтроль та самоаналіз під час розв'язання проблемних завдань, вирішення кейсів тощо), перспективної рефлексії на завершальному етапі (випереджувальне моделювання, прогнозування шляхів власного професійного майбутнього з урахуванням набутого методологічного інструментарію).

Рефлексивну позицію у процесі навчання у ЗВО необхідно також розглядати як своєрідне метапізнання (мислення про власні мисленнєві здібності), що здатне забезпечити критичне ставлення до джерел інформації, теоретичних концепцій та відповідних дослідницьких дій. Також відзначимо, що професійна рефлексія пов'язує методологічні знання з особистісними смислами та цінностями майбутнього спеціаліста (навіщо я обираю цей метод? які етичні наслідки мого професійного вибору?). Стимулювання рефлексії на всіх етапах сприяє тому, що методологічна компетентність стане не формальною вимогою, а

внутрішньою потребою до постійного самовдосконалення, методологічної гнучкості та готовності працювати в умовах професійної невизначеності.

У процесі професійної підготовки майбутніх фахівців з комп'ютерної інженерії доцільно використовувати різні форми та методи розвитку рефлексивної позиції.

По-перше, метод портфоліо (накопичення та публічний аналіз власних програмних продуктів, схемотехнічних рішень та архітектурних проєктів тощо), що дозволяє студенту бачити свій прогрес у динаміці. Так, дослідники Е. Ясинська, І. Навчук [168] розглядають портфоліо-технологію як об'єктивний інструмент обліку діяльності здобувача, орієнтований на кінцевий результат, що забезпечує неперервну оцінку та самооцінку як теоретичної, так і практичної підготовки. На їхню думку, ця форма фіксації індивідуальних і рейтингових досягнень виступає важливим системоутворювальним чинником формування ключових професійних компетентностей майбутніх фахівців. Електронне портфоліо як вискоєфективний засіб для стимулювання мислення в аудиторії, формування стійких навичок самоаналізу, планування навчальної діяльності та розвитку загальної академічної автономії здобувачів досліджують І. Сімерманова та Д. Єславіч [179]. У контексті професійної підготовки фахівців ІТ-сфери Я. Сікора підкреслює можливість наповнення саме е-портфоліо «мультимедійними роботами здобувачів, цифровими документами, зразками програм, програмного коду, вебресурсів тощо, наочними результатами виконання комп'ютерних тестів (графіки, діаграми, інфографіка) тощо» [136, с. 319-320]. *По-друге*, кейс-методи (аналіз реальних критичних збоїв у комп'ютерних системах, вразливостей безпеки й т.п., де студенти мають не просто знайти помилку, а рефлексувати над причинами її виникнення). *По-третє*, технологія взаєморецензування (практика перевірки коду чи схем один одного (в ІТ-сфері – *Code Review*)). *По-четверте*, у межах командної роботи (міні-хакатони, воркшопи) обговорення на кожному етапі сумісної роботи. *По-п'яте*, ведення рефлексивних нотаток під час лабораторних робіт,

де фіксуються не лише проміжні та кінцеві результати, а й хід думок, помилки та шляхи їх подолання (наприклад, у сфері ІТ дебаггінг (від англ. *debugging* виправлення помилок) – процес очищення програм від помилок). Використання цих форм та методів орієнтовано на формування гнучкого інженерного мислення, самоконтролю та готовності до неперервного навчання.

Сутнісні ознаки досліджуваної компетентності, зміст та особливості відповідних педагогічних умов створюють підґрунтя для моделювання процесу формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

У словникових виданнях модель (від лат. *Modus* – міра, фр. *Modele* – зразок) – «у загальному розумінні аналог (графік, схема, знакова система, структура) певного об'єкта (оригіналу), фрагмента реальності, артефактів, витворів культури, концептуально-теоретичних утворень тощо... предметна, знакова чи мислена (уявна) система, що відтворює, імітує або відображає якісь визначальні характеристики» [157, с. 391]; «уявна або матеріально-реалізована система, котра відображає або відтворює об'єкт дослідження (природний чи соціальний) і здатна змінювати його так, що її вивчення дає нову інформацію стосовно цього об'єкта» [33, с. 516].

Відповідно, моделювання – «метод пізнання на основі використання моделі як засобу дослідження явищ і процесів природи» [107, с.26]. Для процесу моделювання у педагогічних дослідженнях суттєвим є зауваження С. Сисоєвої та Т. Кристопчук про те, що «що будь-яка модель завжди бідніша за прототип, що вона відображає лише його окремі сторони і зв'язки, оскільки теоретичне моделювання завжди містить абстрагування» [131, с. 102].

На Рис. 2.1. представлено структурно-функційну модель процесу формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, що містить цільовий, теоретико-методологічний, організаційно-змістовний та результативний блоки.

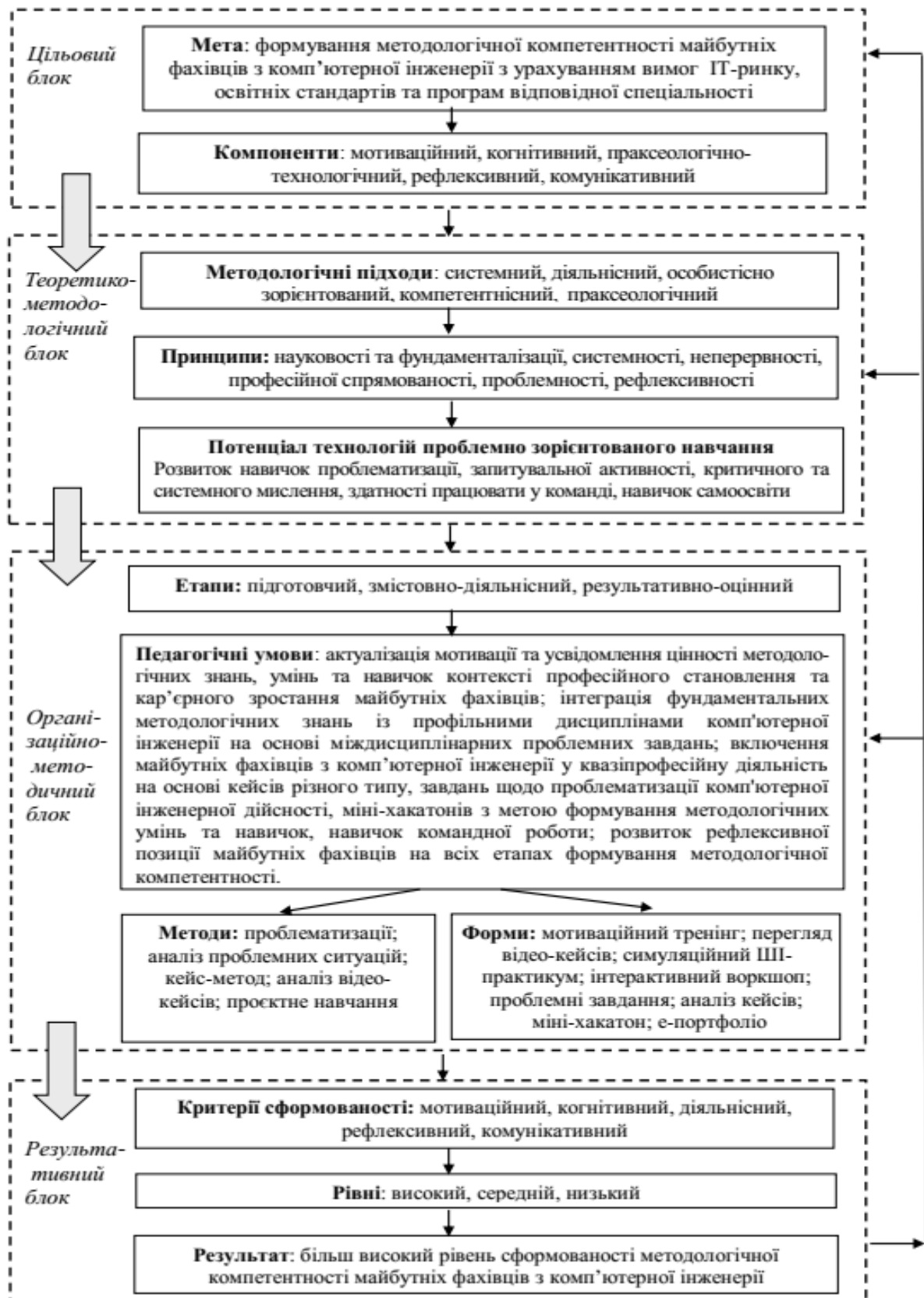


Рис. 2.1. Структурно-функційна модель формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії

Першим складником виступає цільовий блок, який окреслює підґрунтя усього процесу формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії з урахуванням актуальних вимог ІТ-ринку, освітніх стандартів та відповідних спеціальних програм. Цей блок виокремлює мотиваційний, когнітивний, праксеологічно-технологічний, рефлексивний та комунікативний компоненти, а також репрезентує відповідні педагогічні умови. Серед цих умов – актуалізація мотивації й усвідомлення цінності методологічних знань та умінь для кар'єрного зростання, інтеграція фундаментальних методологічних знань із профільними дисциплінами через міждисциплінарні проблемні завдання, залучення студентів до квазіпрофесійної діяльності на основі кейсів й міні-хакатонів, а також постійне стимулювання рефлексивної позиції.

Наступним є теоретико-методологічний блок, який визначає науковий фундамент та орієнтири освітнього процесу та ґрунтується на системному, діяльнісному, особистісно зорієнтованому, компетентнісному та праксеологічному методологічних підходах. Реалізація цих підходів відбувається через дотримання принципів науковості та фундаменталізації, системності, неперервності, професійної спрямованості, проблемності та рефлексивності.

Принципи науковості та фундаменталізації вимагають побудову навчального контенту на основі доведених наукових фактів, теоретичних моделей та академічних методів досліджень, що формує у здобувачів науковий світогляд, теоретичне підґрунтя для навчально-пізнавальної, квазіпрофесійної та майбутньої професійної діяльності у галузі комп'ютерної інженерії. Принцип системності передбачає формування методологічної компетентності у взаємозв'язку та взаємозумовленості всіх компонентів, логічне структурування освітнього процесу на основі доповнюваності форм та методів, урахування особливостей ІТ-освіти, вимог ринку праці, тенденцій розвитку галузі. Принцип неперервності забезпечує наскрізний характер та поетапність формування методологічних знань, умінь та навичок, зумовлює формування умінь та навичок навчання упродовж життя. Принцип

професійної спрямованості підпорядковує зміст, методи та засоби навчання вимогам ІТ-галузі, моделює контекст майбутньої діяльності безпосередньо в навчальному середовищі. Принцип проблемності орієнтує організацію освітнього процесу навколо критичних та відкритих питань через використання методів проблемно зорієнтованого навчання, дослідницьких завдань, що стимулюють активну пізнавальну діяльність. Принцип рефлексивності пов'язано із інтеграцією в освітній процес форм та методів самоаналізу, взаємооцінювання, зворотного зв'язку, що спонукає здобувачів вищої освіти до критичного ставлення до власного процесу особистісного та професійного розвитку.

Теоретико-методологічний блок репрезентує потенціал технологій проблемно зорієнтованого навчання, який спрямований на розвиток навичок проблематизації, запитувальної активності, критичного й системного мислення, а також здатності працювати в команді та ефективно навчатися самостійно.

Організаційно-методичний блок забезпечує практичне втілення розробленої моделі на основі відповідних педагогічних умов. Поетапна реалізація здійснюється через підготовчий, змістовно-діяльнісний і результативно-оцінний етапи. Освітній процес насичується методами проблематизації, аналізу проблемних ситуацій, кейс-методом, аналізом відео-кейсів та проєктним навчанням. Для формування методологічної компетентності здобувачів спеціальності «Комп'ютерна інженерія» застосовуються різноманітні форми роботи, зокрема мотиваційні тренінги, перегляд відео-кейсів, симуляційні ІІІ-практикуми, інтерактивні воркшопи, розв'язання проблемних завдань, аналіз кейсів, проведення міні-хакатонів та формування е-портфоліо. Завершальним елементом системи є результативний блок, який підсумовує ефективність запроваджених заходів. Оцінювання здійснюється за мотиваційним, когнітивним, діяльнісним, рефлексивним та комунікативним критеріями. Кінцевим результатом функціонування усієї моделі є досягнення більш високого рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Отже, відповідно до завдань дослідження, педагогічні умови трактуються як цілеспрямований відбір, проєктування та інтеграція змісту, форм і методів проблемно зорієнтованого навчання з метою формування в майбутніх фахівців з комп'ютерної інженерії методологічної компетентності. Реалізація цих умов дозволяє створити таке освітнє середовище, у якому академічні завдання природно трансформуються у вирішення реальних інженерних кейсів. Такий підхід суттєво інтенсифікує освітній процес, стимулює становлення проактивної особистості майбутнього фахівця з комп'ютерної інженерії, який володіє розвиненим алгоритмічним мисленням, методологічними вміннями та навичками та здатністю оперативно адаптуватися до динамічних вимог сучасної індустрії високих технологій.

2.2. Упровадження педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії

Дослідно-експериментальна робота з формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії була організована з урахуванням теоретичних засад дослідження, представлених у підрозділі 2.1. педагогічних умов: актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності.

Важливо підкреслити, що визначені педагогічні умови впроваджувалися комплексно та взаємозумовлено з використанням форм та методів проблемно зорієнтованого навчання.

Дослідно-експериментальна робота проводилася на базі Національного університету біоресурсів і природокористування України зі здобувачами вищої освіти спеціальності 123 «Комп'ютерна інженерія» упродовж 2023-2026 рр. та включала такі етапи: підготовчий, змістовно-розвивальний, підсумково-прогностичний.

На підготовчому етапі на основі відповідного інструментарію було здійснено діагностику сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії; підготовлено навчально-методичне забезпечення формульованого етапу дослідження. Наступний – змістовно-розвивальний етап – пов'язано із упровадженням педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії. Підсумково-прогностичний етап передбачав проведення підсумкової діагностики сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії, визначення студентами подальших перспектив професійного зростання.

Реалізація першої умови – актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців – передбачала додатковий акцент у процесі аудиторної та позааудиторної роботи на значущості методологічного забезпечення професійної діяльності, актуальності методологічних знань та умінь у процесі виконання функцій фахівця з комп'ютерної інженерії.

З метою формування мотивації студентів-першокурсників до оволодіння методологічними знаннями та уміннями як складника майбутньої професійної діяльності розроблено та проведено позааудиторний **інтерактивний воркшоп «Ринок праці очима першокурсника: де я зараз і де хочу бути»** (Додаток В.1). Суть заходу полягала у створенні проблемно зорієнтованого середовища, де студенти першого курсу самостійно досліджували актуальні вакансії (рівня Junior/Trainee) у сфері комп'ютерної інженерії на провідних платформах працевлаштування (DOU, Djinni, Work.ua та ін.). Замість традиційного пасивного слухання лекцій про кар'єру, студенти працювали в міні-групах, виконуючи роль

аналітиків, декомпонували вимоги роботодавців та шукали прямі зв'язки між цими вимогами і робочим навчальним планом своєї спеціальності.

Мета воркшопу – подолати розрив між абстрактними уявленнями першокурсників про професію та реальними вимогами ІТ-індустрії, а також трансформувати зовнішню (оцінювальну) мотивацію студентів у внутрішню (пізнавальну-професійну) через усвідомлення практичної цінності фундаментальних дисциплін. Першокурсники часто приходять до університету з хибним уявленням, що інженерія зводиться лише до вивчення мов програмування, через що знецінюють фундаментальні дисципліни (вищу математику, фізику, теорію алгоритмів, архітектуру ЕОМ). Занурення студентів у реальний контекст ринку праці ініціює когнітивний дисонанс: вони бачать, що роботодавці вимагають не просто знання синтаксису коду, а навичок системного мислення, розуміння архітектури мереж, умінь проєктувати бази даних та оптимізувати процеси. Цей воркшоп виступає «стартовою точкою», яка формує запит студента на методологічне знання. Коли студент самостійно знаходить у вимогах до DevOps-інженера чи фахівця з кібербезпеки розуміння принципів маршрутизації чи криптографії, у нього зникає питання «навіщо ми це вчимо?». Таким чином, навчальні дисципліни перестають бути абстрактними і стають усвідомленими схождениями до професійної мети.

Під час воркшопу студентам пропонувалися питання, спрямовані на рефлексивну оцінку отриманої інформації та побудову подальших перспектив професійного розвитку: Які «м'які навички» (Soft Skills) роботодавці вимагають від фахівців технічного профілю? У межах якого освітнього компоненту ви можете потренувати навички командної роботи та тайм-менеджменту, які так активно шукають роботодавці? Де я знаходжуся зараз (моя точка А) у порівнянні з вимогами до Junior-спеціаліста? та ін.

Для оцінки впливу заходу на мотиваційну сферу студентів було зібрано їхні рефлексивні відгуки. Наведемо декілька фрагментів, що демонструють якісні зміни у ставленні здобувачів до освітнього процесу:

Денис З. *«Найбільшим інсайтом для мене стало те, що ринок праці шукає людей, які вміють вирішувати проблеми (problem-solving), а не просто писати код за готовим шаблоном. Ми аналізували вимоги до мережесевих інженерів, і там скрізь наголос на проектуванні безпеки, вмінні читати документацію (англійською!) та аналізувати збої. Цей захід показав мені мої слабкі місця. Я зрозумів, що мені треба не просто "здати лабораторну і забути", а намагатися зрозуміти саму логіку побудови мереж і систем. Це зовсім інший рівень відповідальності».*

Максим Л. *«Мене часто дратували предмети, які, як мені здавалося, "не стосуються спеціальності". Але під час виконання завдання ми мали співставити вимоги вакансій з нашими предметами. Коли ми шукали компетенції для фахівців з хмарних технологій, то виявилось, що для розуміння навантажень і оптимізації потрібні знання з системного аналізу і навіть базової статистики. Воркшоп ніби зняв з мене "рожеві окуляри". Я склав для себе чіткий план, на які дисципліни зараз треба кинути всі сили, щоб через три роки відповідати вимогам реальних компаній».*

Проведений аналіз рефлексій доводить, що залучення студентів до роботи з реальними ринковими даними успішно активізує внутрішню мотивацію. У здобувачів освіти формується первинне розуміння того, що методологічна компетентність (вміння системно мислити, проектувати та аналізувати тощо) є базовим критерієм їхньої майбутньої професійної конкурентоспроможності. Доцільно проводити захід двічі на 1 курсі та на 3 курсі з метою фіксації динаміки розуміння студентами ринку праці та власної професійної ідентичності як індикатора сформованості методологічної компетентності.

Відповідно до першої педагогічної умови зі студентами 3-го курсу спеціальності «Комп'ютерна інженерія» в НУБІП було проведено мотиваційний тренінг **«Керуй алгоритмом: мистецтво методологічного мислення в ІТ»** (Додаток В.1), побудований на основі проблемних ситуацій за моделлю 4C/ID (*Four-Component Instructional Design*) [221].

Метою заходу було зміщення фокусу здобувачів освіти з простого інструментального використання команд (налаштування обладнання) на формування системного мислення, навичок проблематизації та запитувальної активності в процесі пошуку та усунення несправностей (*Troubleshooting*).

Заняття розпочалося зі створення ситуації «когнітивного конфлікту». Студенти були об'єднані в міні-групи, кожна з яких отримала опис критичної мережевої інцидент-заявки (тикета) з неповними та заплутаними вихідними даними. На першому етапі здобувачам свідомо не надавалися інструкції, що спровокувало інтуїтивний, хаотичний пошук несправностей. Після того, як більшість груп зайшла у глухий кут, викладач презентував три базові алгоритми методологічного мислення інженера (Bottom-Up, Top-Down, Divide and Conquer) на основі еталонної моделі OSI. Наступний етап вимагав від студентів застосування виключно алгоритмізованого підходу. Наведемо декілька фрагментів рефлексивних відгуків студентів:

Вячеслав Б. зауважив про відсутність системного підходу на початку викликала роздратування в команді: «Кожен пропонував свою команду наосліп. Тільки після складання покрокового чек-листа ми додумалися перевірити статус Spanning Tree Protocol і побачили помилку консистенції портів».

Дмитро В. підкреслив, що цей кейс довів йому марність знання синтаксису команд без розуміння архітектури: «Можна знати напам'ять усі команди OSPF, але вони абсолютно непотрібні, якщо ти не вмієш діагностувати проблему на рівні L2. Методологія змушує мислити послідовно».

Сергій Л. зауважив, що тренінг змінив його ставлення до професії: «До сьогодні я думав, що крутий інженер – це той, хто швидко друкує команди в консолі без підказок. Тепер я розумію, що крутий інженер – це той, хто вміє правильно структурувати пошук проблеми, навіть якщо для самої команди йому знадобиться документація».

Аналіз рефлексивних суджень студентів підтверджує значущість реалізації педагогічної умови, пов'язаної із мотивацією. У здобувачів було зафіксовано

чітке усвідомлення обмеженості суто інструментальних навичок і формування внутрішньої потреби в опануванні методологічними знаннями та вміннями як основи їхньої майбутньої професійної діяльності в галузі комп'ютерної інженерії.

Одним із дієвих інструментів проблемно зорієнтованого навчання є **використання відео-кейсів** як унаочнення проблемного бачення дійсності. З метод підсилення мотивації студентів до оволодіння загальнонауковими та специфічними методами у галузі комп'ютерної інженерії було запропоновано проаналізувати фрагменти виступів провідних експертів галузі на конференціях TEDx (TED – некомерційна організація в США, що займається популяризацією проривних технологій та ідей у різних сферах). Відеозаписи на сайті TED.com мають мільярдні перегляди. Так, під час вивчення дисциплін циклу професійної підготовки (наприклад, «Архітектура комп'ютерів», «Технології проектування цифрових систем») на початку лекції ми пропонували студентам переглянути фрагмент виступу Д. Пневматикатоса на конференції TEDx «*A brief look at the history of Computer Architecture*» (Короткий погляд на історію комп'ютерної архітектури) (Pnevmatikatos D. A Brief Look at the History of Computer Architecture. *ted.com*. URL: https://www.ted.com/talks/dionisios_pnevmatikatos_a_brief_look_at_the_history_of_computer_architecture).

Після перегляду відеоматеріалу студентам пропонується низка проблемних запитань, що мають на меті змістити фокус їхньої уваги з суто апаратних характеристик на необхідність розвитку системного мислення, розуміння еволюційних закономірностей розвитку технологій та усвідомлення ролі сучасного інженера. Зокрема такі запитання: 1) Чому сьогодні розробнику (як апаратного, так і програмного забезпечення) вже недостатньо знати лише свою вузьку ділянку роботи, а критично необхідно розуміти архітектуру в цілому? 2) Яким чином системне мислення допомагає сучасному комп'ютерному інженеру перетворювати фізичні обмеження (як-от межі закону Мура) на нові інноваційні рішення, а не сприймати їх як глухий кут? 3) Зважаючи на стрімку еволюцію обчислювальних систем, про яку говорить доповідач, які складники професійної компетентності, окрім знання актуальних сьогодні мов

програмування чи схемотехніки, дозволять вам залишатися затребуваним фахівцем через 5–10 років?

Обговорення цих запитань стимулювало майбутніх фахівців мислити категоріями складних систем і процесів. Під час застосування цієї методики ми фіксували та аналізували відповіді здобувачів освіти. Наведемо типові варіанти відповідей студентів: *Володимир Т. зазначив, що «коли інженери зіткнулися з проблемою виділення тепла, вони не здалися, а змінили саму парадигму – перейшли до багатоядерності та спеціалізованих прискорювачів (FPGA, GPU). Це означає, що сучасний інженер має бути не просто виконавцем за інструкцією, а дослідником. Системне мислення потрібне для того, щоб вміти вийти за рамки звичного підходу, проаналізувати проблему з різних боків і знайти нестандартне архітектурне рішення».* *Іван Д. зауважив, що «головна ідея в тому, що технології неминуче змінюються і старіють. Ті технічні інструменти, які ми вчимо зараз, можуть зникнути за кілька років. Щоб залишатися професіоналом, потрібно розуміти самі принципи розвитку технологій, вміти адаптуватися та швидко вчитися новому. Для цього потрібна широка ерудиція, здатність аналізувати тренди та розуміти, як працює індустрія в комплексі, а не лише паяти плати чи писати скрипти».*

На вступних лекціях з дисципліни «3D моделювання і друк» фрагмент виступу Бенджаміна Ділленбургера «Еволюція 3D-друку» (Dillenburger B. 3D Print Evolution. *ted.com*. URL: https://www.ted.com/talks/benjamin_dillenburger_3d_print_evolution) допомагає розкрити парадигмальний зсув: перехід від традиційного проєктування до алгоритмічної генерації фізичних об'єктів («кодування архітектури»), де цифрові дані матеріалізуються у складні просторові форми, а технологічна складність виробництва стає «безкоштовною» завдяки обчислювальним потужностям. Мета демонстрації цього відео-кейсу полягала не у вивченні принципів роботи 3D-принтера, а у проблематизації ролі комп'ютерного інженера в сучасному світі. Після перегляду студентам було запропоновано відповісти на низку запитань, орієнтованих на актуалізацію системного мислення та методологічної рефлексії: Доповідач зазначає, що

архітектуру тепер не креслять, а «кодують», і дані безпосередньо стають матеріалом. Як ця зміна парадигми впливає на рівень відповідальності комп'ютерного інженера? Чому сьогодні недостатньо просто написати код без помилок, а критично необхідно розуміти фізичні наслідки його виконання? Проєкт зі створення цілої кімнати за допомогою алгоритмів вимагав співпраці програмістів, архітекторів, матеріалознавців та інженерів-механіків. Як системне мислення дозволяє ІТ-фахівцю знайти спільну мову з представниками нетехнічних галузей для вирішення такої комплексної проблеми?

Впровадження такого формату роботи дозволило перевести мислення студентів із площини суто технічного виконання алгоритмів у площину системного аналізу наслідків цифровізації фізичного світу. Наведемо відгуки студентів:

Богдан С.: «Це відео змушує по-іншому поглянути на наш код. Якщо раніше помилка в програмі або неправильно спроектована база даних призводили лише до зависання екрану або втрати інформації, то тепер наш код будує фізичні об'єкти. Алгоритмічна помилка у генеративному дизайні може призвести до того, що надрукована конструкція обвалиться. Тому сучасний інженер має мислити системно, повинен розуміти закони фізики, супротив матеріалів і те, як цифрові дані перетворюються на реальний світ». Владислав П.: «Використання обчислювальних потужностей для 3D-друку дозволяє створювати пористі, легкі, але дуже міцні конструкції, як кістки в тілі людини. Це економить тони матеріалу. Тобто комп'ютерна інженерія сьогодні безпосередньо впливає на екологію та економіку. Системне мислення полягає в тому, щоб бачити цей ланцюжок: від оптимізації алгоритму структури даних до збереження природних ресурсів під час фізичного будівництва».

У процесі використання відео-кейсів для нас важливо було окрім суто предметного змісту унаочнити стиль виступів відомих фахівців ІТ-галузі, логіку аргументації, що віддзеркалює специфіку інженерного мислення, інноваційність професійного пошуку. Отже, звернення до автентичних проблемно зорієнтованих відео-кейсів на початку лекційних занять виконує роль потужного

когнітивного тригера. Це перетворює їх із пасивних споживачів інформації на активних суб'єктів інженерної творчості, здатних вирішувати нетипові, комплексні завдання в умовах стрімкої еволюції цифрових технологій.

В умовах тотального поширення штучного інтелекту студенти виявили особливий інтерес до такого інноваційного дидактичного інструменту як **симуляційний ІІІ-практикум «Інженерне мислення: точка зламу»** (Додаток В.3). Суть даного практикуму полягає у створенні контрольованого проблемного середовища, де студент вступає в інтерактивний діалог із генеративним штучним інтелектом, налаштованим на виконання ролі висококваліфікованого фахівця (Senior Network Architect / СТО). Практикум базується на ідеях М. Севін-Бейден (M. Savin-Baden), яка обґрунтувала модель PBL для критичної дискусійності (критичної компетентності) [209], в тому числі із використанням методу Сократа.

Головна мета симуляції полягає не в перевірці декларативних знань чи завченого синтаксису команд (рівня виконавця), а в ініціації когнітивного конфлікту. Студент свідомо провокується на захист спрощених, суто ремісничих рішень (*«головне – налаштувати обладнання, щоб воно запрацювало»*). У відповідь ІІІ-співрозмовник, оперуючи реальними кейсами інженерних збоїв та вразливостей, розхитує цю позицію, демонструючи катастрофічні наслідки відсутності системного підходу. Таким чином, відбувається злам стереотипного мислення студента. Майбутній фахівець з комп'ютерної інженерії на власному (симульованому) досвіді доходить висновку, що успішна професійна діяльність неможлива без оволодіння методологічними знаннями: архітектурним плануванням, моделюванням загроз, принципами відмовостійкості та масштабованості. Це забезпечує перехід від зовнішньої мотивації (*«вивчити заради оцінки»*) до внутрішньої (*«зрозуміти методологію, щоб не допустити краху системи в реальній роботі»*). Фрагменти протоколів діалогів під час ІІІ-практикуму представлено у Додатку В.3.

У процесі експериментальної роботи значна увага була приділена інтеграції фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань. Для

цього було розроблено та впроваджено комплекс наскрізних проблемно зорієнтованих сценаріїв, які вимагали від здобувачів вищої освіти одночасного залучення теоретичного апарату фундаментальних наук та практичного інструментарію прикладних ІТ-дисциплін.

Реалізація поставленого завдання розпочалася з удосконалення змісту лабораторних, практичних та позааудиторних занять. Відповідно до теоретичних висновків першого розділу щодо необхідності подолання вузькодисциплінарної фрагментарності та розвитку системного інженерного мислення, традиційні завдання, які передбачали лінійне виконання покрокових інструкцій, були доповнені комплексними проблемно зорієнтованими інженерними викликами з високим ступенем невизначеності. Базовим освітнім компонентом для інтеграції виступив ОК «Комп'ютерні мережі» у тісному міждисциплінарному взаємозв'язку з іншими фаховими дисциплінами.

Одним із таких сценаріїв стало завдання з проєктування захищеної корпоративної системи збору та аналізу мережевої статистики. Студентським командам було запропоновано розробити архітектуру, де інфраструктурний рівень налаштовувався в межах ОК «Комп'ютерні мережі (частина 2)» під час виконання Лабораторної роботи №7 «Комплексне завдання щодо розгортання ACL для IPv4». Водночас бекенд для обробки зібраної телеметрії та візуалізації даних реалізовувався засобами Python і вимагав у здобувачів знань із «Кросплатформне програмування (Python)». Здобувачі стикалися з необхідністю узгодження політик контролю доступу мережевого рівня із логікою роботи вебзастосунку, що перетворювало їхні знання на реальний інструмент розв'язання комплексної проблеми.

Наступним етапом стало занурення майбутніх фахівців у середовище критичних інфраструктурних збоїв. У межах опанування ОК «Комп'ютерні мережі (частина 2)» (Тема «Принцип роботи протоколу STP», Лабораторна робота №2 «Впровадження VLAN та транкових каналів») студентам пропонувалося розгорнути комутовану мережу підприємства. Під час виконання цього проблемного завдання була змодельована ситуація технологічної кризи –

широкомовний шторм (Broadcast Storm) внаслідок утворення L2-петлі та відмови протоколу STP. Для подолання суперечності здобувачі вищої освіти мусили зупинити процес хаотичного підбору команд і перейти до системної декомпозиції проблеми (алгоритм «Bottom-Up»), самостійно локалізавши джерело шторму в конфігурації VLAN. Цей прецедент став потужним каталізатором розвитку методологічної компетентності на рівні системного аналітика.

Інший приклад міждисциплінарного синтезу полягав у тому, що для діагностики вразливостей, аналізу PCAP-дамів у Wireshark та оптимізації конфігурацій здобувачі активно використовували знання з ОК «Основи пром프트-інжинірингу» (зокрема з тем «Розробка мультикрокових пром프트-систем» та «Створення AI-асистента»). Студенти вчилися формувати точні алгоритмічні запити для великих мовних моделей, що вимагало від них глибокої формалізації мережевого контексту та рефлексивного критичного аналізу згенерованих штучним інтелектом конфігураційних рішень. Це дозволило органічно поєднати практику конструювання високонавантажених комп'ютерних мереж з новітніми технологіями автоматизації інженерного пошуку, закріплюючи методологічну компетентність на рівні преадаптивності фахівця.

Реалізація наскрізних проблемно зорієнтованих сценаріїв зумовила суттєву зміну ролі викладача, відмову від функції репродуктивних трансляторів правильних відповідей, набуття досвіду фасилітаторів та наставників. Коли студентські команди заходили у глухий кут під час проєктування телеметрії, налаштування хмарних політик чи аналізу управління пам'яттю, викладач не вказував на синтаксичну помилку, а за допомогою системи навідних запитань спонукав студентів до перевірки їхніх архітектурних рішень. Відбувалася стимуляція критичного мислення: студентів просили обґрунтувати, чому обрано саме такий криптографічний алгоритм, як саме операційна система виділяє ресурси під цей процес, або як зміна одного параметра в JSON-файлі вплине на загальний життєвий цикл екосистеми. Така форма взаємодії змушувала майбутніх фахівців постійно утримувати у фокусі уваги весь

комплекс міждисциплінарних зв'язків, не дозволяючи їм утверджуватися на позиціях вузькотехнічного ремісництва.

У процесі моніторингу вирішення студентами комплексних завдань ми фіксували суттєве скорочення часу на адаптацію до нових нетипових умов. Студенти почали демонструвати здатність до екстраполяції набутих методологічних навичок: навички системного аналізу, здобуті під час дослідження управління пам'яттю у Python, успішно переносилися ними на оптимізацію мережевого трафіку чи налаштування хмарної інфраструктури. Здобувачі вищої освіти припинили сприймати фундаментальні математичні чи логічні дисципліни як відірвану від реальності теорію; ці знання інтеріоризувалися, ставши невід'ємною частиною їхнього повсякденного інженерного інструментарію.

Відповідно до завдань дослідження особливу увагу було приділено формуванню умінь та навичок студентів щодо використання загальнонаукових методів пізнання (аналіз, синтез, класифікація, порівняння та ін.). У межах освітніх компонентів фахової підготовки майбутніх фахівців з комп'ютерної інженерії «Комп'ютерні мережі» та ін. було розроблено та впроваджено **комплекс завдань, які були безпосередньо пов'язані із формуванням навичок застосування загальнонаукових методів у процесі вирішення професійних ситуацій.** З метою актуалізації знань студентів щодо сутнісних характеристик загальнонаукових методів пізнання було розроблено відповідні алгоритмічні приписи (правила) (Додаток В.4).

Важливо підкреслити, що зазначені приписи (правила) стосуються саме професійного контексту комп'ютерної інженерії. Акцентовано увагу на типології аналізу в інженерній діяльності (проблемний, структурний, системний, причинно-наслідковий, праксеологічний, аксіологічний, ситуаційний, програмно-цільовий), модифікаціях синтезу (структурний, архітектурно-системний, оптимізаційний, інтеграційний). Зазначено потенціал кожного загальнонаукового методу для успішного вирішення відповідних професійних завдань.

Для самостійної роботи здобувачів щодо загальнонаукових методів було запропоновано звернутися до словників та посібників: Гладун А. Я., Пучков О. О.,

Субач І. Ю., Хала К. О. Англо-український словник термінів з інформаційних технологій та кібербезпеки (2018) [23], Голибард Є. Основи технології системного мислення : підручник (2017) [24]; Максименко С. Д., Меєрович М. Й., Шрагіна Л. І. Системне мислення: формування і розвиток : навч. посіб. (2020) [81]; Основи наукових досліджень. Конспект лекцій: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія» (2022) [107] та ін.

У Табл. 2.3. представлено типологію завдань у межах конкретних освітніх компонентів, які спрямовані на усвідомлення сутності та контексту використання різних загальнонаукових методів.

Таблиця 2.3.

Формування навичок використання загальнонаукових методів пізнання у професійній підготовці майбутніх фахівців з комп'ютерної інженерії

Освітній компонент	Приклади завдань
Аналіз (від грец. <i>analysis</i> – розкладення, розчленування) – мисленнєва операція розподілу об'єкту на складові частини або характеристики. Здійснюється шляхом виділення в об'єкті істотних властивостей, ознак, зв'язків, стосунків	
ОК Програмування (1 курс)	Розбиття неструктурованого монолітного коду (наприклад, мовою C) на окремі логічні функції з визначенням областей видимості локальних та глобальних змінних.
ОК Комп'ютерні мережі (3-4 курс)	Декомпозиція перехопленого кадру Ethernet: виокремлення MAC-адрес, розбір IP-заголовка та корисного навантаження для розуміння багаторівневої інкапсуляції даних. Зіставлення протоколів динамічної маршрутизації (наприклад, RIP та OSPF) для визначення оптимального варіанту масштабування корпоративної мережі. Розбиття загального пулу IP-адрес підприємства на логічні підмережі (створення VLAN) відповідно до структури відділів (бухгалтерія, IT, гостьова мережа).
Синтез – це уявне об'єднання частин, властивостей, дій в єдине ціле. Створення нових систем. Синтез відноситься до основної мисленнєвої операції.	
ОК Програмування (1 курс)	Побудова детальної блок-схеми (або написання псевдокоду) для складного алгоритму перед початком безпосереднього написання синтаксичного коду мовою C.
ОК Комп'ютерні мережі (3-4 курс)	Проектування топології корпоративної мережі підприємства з нуля: вибір кількості маршрутизаторів, комутаторів, розбиття на VLAN та розподіл IP-адрес на основі технічного завдання замовника. Інтеграція існуючої локальної дротової мережі (LAN), нового бездротового сегмента (WLAN) та демілітаризованої зони (DMZ) у єдину, захищену та маршрутизовану інфраструктуру.
Класифікація (від лат. <i>classis</i> – розряд, клас та <i>facio</i> – роблю, розкладаю) – мисленнєва операція, спрямована на розподіл предметів, явищ і понять за видами, ознаками, класами,	

групами, розрядами залежно від їх істотних ознак в групі, підгрупі, класі – кінцева або нескінченна сукупність виділених за деякою ознакою предметів, що розуміється як ціле)	
ОК Програмування (1 курс)	Групування розрізнених неструктурованих змінних у логічні структури даних (використання struct у мові C) для системного опису складної сутності (наприклад, сутності «Студент» чи «Товар»).
	Вибір найефективнішого алгоритму сортування (із класу квадратичних або логарифмічних) під специфічні умови: обмежений обсяг оперативної пам'яті або частково відсортований вхідний масив.
ОК Комп'ютерні мережі (3-4 курс)	Аналіз перехопленого мережевого трафіку (pcap-файлу) для ідентифікації типу кібератаки (наприклад, класифікація інциденту як DoS-атаки, ARP-spoofing чи сканування портів).
	Вибір оптимального протоколу маршрутизації (із класу дистанційно-векторних чи каналу зв'язку) або типу середовища передачі (мідь/оптика) під конкретні вимоги масштабу та бюджету.
	Розробка студентом власної ієрархії політик доступу (ACL) для маршрутизаторів корпоративної мережі.
Порівняння – мисленнєва операція, при якій встановлюються схожість і відмінності між об'єктами. Основою для порівняння бувають, як правило, істотні ознаки порівнюваних об'єктів. Порівнюючи явища одне з одним, відзначається як схожість, так і відмінність їх у певних стосунках, їх тотожність або протилежність	
ОК Програмування (1 курс)	Обґрунтований вибір між використанням рекурсії та ітерації (циклів) для розв'язання математичної задачі з урахуванням жорстких обмежень на обсяг доступної стекової пам'яті мікроконтролера.
ОК Комп'ютерні мережі (3-4 курс)	Вибір протоколу транспортного рівня (TCP чи UDP) для трансляції потокового відео з камер спостереження з урахуванням жорстких обмежень щодо затримок сигналу та допустимості втрати кадрів.
	Виявлення принципових відмінностей у логіці роботи комутатора (Switch) та маршрутизатора (Router): чому зовні схожі пристрої працюють на різних рівнях моделі OSI та використовують різні типи адрес (MAC проти IP). Побудова у симуляторі двох однакових топологій: однієї зі статичною маршрутизацією, іншої – з динамічною (OSPF). Замір часу відновлення зв'язку (часу збіжності) після імітації обриву головного кабелю.
	Порівняння структури заголовка і можливостей протоколу IPv4 з сучасним IPv6 для розуміння причин еволюції: як вичерпання адресного простору та потреба у вбудованій безпеці (IPsec) змінили технологію.
Узагальнення – мисленнєва операція, при якій відбувається отримання висновку на основі уявного об'єднання предметів і явищ за їх загальними та істотними ознаками	
ОК Програмування (1 курс)	Замір часу виконання алгоритму для різних розмірів вхідного масиву ($N=10, 100, 1000, 10000$) та індуктивне виведення загальної закономірності щодо його обчислювальної складності (наприклад, емпіричне доведення того, що алгоритм має квадратичну складність).
ОК Комп'ютерні мережі (3-4 курс)	Розробка єдиного універсального скрипта (шаблону) для базового налаштування комутаторів (VLAN, безпека портів, паролі), який можна застосувати до будь-якого нового пристрою в мережі, замість ручного конфігурування кожного окремо.
	Проведення серії стрес-тестів у симуляторі з поступовим збільшенням обсягу трафіку та самостійне виведення емпіричного правила щодо межі пропускної здатності каналу і моменту початку відкидання пакетів (drop).
	Перенесення загальних принципів ієрархічної адресації та маршрути-

	зації, засвоєних під час вивчення протоколу IPv4, на розв'язання задачі з розбиття на підмережі в абсолютно новому стандарті IPv6. творення студентом єдиного, структурованого протоколу (чек-листа) з діагностики мережі (наприклад, "Алгоритм дій при втраті зв'язку між VLAN") на основі хаотичного досвіду помилок, отриманого під час виконання попередніх лабораторних робіт.
Абстрагування (від лат. <i>abstractio</i> – відволікання) – це мисленнєва операція, яка виділяє окремі ознаки, елементи та відділяє їх від інших і від самих об'єктів.	
ОК Програмування (1 курс)	Створення концептуальної структури даних (наприклад, UML-діаграми чи ER-діаграми) на основі текстового опису предметної області, відкидаючи несуттєві атрибути (наприклад, при моделюванні розкладу занять абстрагуємося від кольору стін в аудиторії).
ОК Комп'ютерні мережі (3-4 курс)	Аналіз бізнес-вимог замовника (наприклад, "потрібно об'єднати два офіси, щоб бухгалтерія не бачила мережу розробників") та створення на їх основі абстрактного графа мережі (із зазначенням логічних вузлів, тунелів та підмереж), ігноруючи моделі конкретного обладнання чи довжину кабелів. Проектування загальної абстрактної моделі безпеки (політики доступу) на рівні підприємства у вигляді набору логічних правил (хто куди має доступ), залишаючи технічну реалізацію (ACL, Firewall rules) на нижчому рівні. Налаштування статичних маршрутів на 3-му рівні моделі OSI (мережевому), свідомо абстрагуючись від фізичної реалізації каналів (мідь, оптика чи радіо) та MAC-адрес (2-й рівень). Розрахунок теоретичної пропускної здатності та оптимальних шляхів у мережі на основі алгоритму Дейкстри, ідеалізуючи середовище та не враховуючи реальні затримки на комутаторах, колізії чи відсоток пошкоджених пакетів.
Дедукція (лат. <i>deductio</i> , від <i>deduco</i> – «низводжу, відводжу») виведення часткових наслідків із загальних передумов; застосування загальних законів, правил, стандартів або алгоритмів до конкретних інженерних ситуацій	
ОК Програмування (1 курс)	Спираючись на загальне правило мови C («індексація масивів починається з 0 і закінчується N-1»), логічно локалізувати конкретний рядок коду, де виникає помилка «вихід за межі масиву» (наприклад, у циклі з умовою $i \leq N$).
ОК Комп'ютерні мережі (3-4 курс)	Використовуючи загальне правило моделі OSI («маршрутизатор відкидає пакети з TTL=0»), логічно вирахувати та знайти конкретний вузол, на якому утворилася петля маршрутизації (routing loop), аналізуючи вивід команди traceroute. Застосування загальних правил роботи протоколу STP (Spanning Tree Protocol) до специфічної фізичної топології замовника для логічного вирахування того, які саме порти комутаторів будуть заблоковані для уникнення широкомовних штормів. Знаючи загальні механізми роботи протоколу TCP (вікно ковзання, контроль перевантаження), логічно спрогнозувати точну поведінку мережі та падіння швидкості передачі, якщо в симуляторі штучно встановити затримку (delay) каналу у 500 мс. Використовуючи загальні знання про вразливості протоколу ARP (відсутність автентифікації), логічно спланувати та провести атаку ARP-spoofing на конкретну віртуальну топологію для перехоплення незашифрованого пароля адміністратора.

Індукція (лат. <i>inductio</i> «привести (до)») – метод пізнання, за якого загальний висновок про властивості множини елементів або явищ робиться на основі дослідження окремих (часткових) фактів, експериментів чи спостережень	
ОК Програмування (1 курс)	Запуск власноруч написаного алгоритму (наприклад, пошуку підрядка) на текстах різного обсягу (10Кб, 100Кб, 1Мб), замір часу виконання та індуктивне виведення математичної залежності (обчислювальної складності) цього алгоритму.
ОК Комп'ютерні мережі (3-4 курс)	Проведення серії експериментів у симуляторі зі зміною розміру MTU (Maximum Transmission Unit) та самостійне виведення загальної закономірності щодо впливу фрагментації на затримки та загальну пропускну здатність каналу. Дослідження невідомого мережевого вузла: відправлення пакетів на різні порти (TCP/UDP) та з різними прапорцями, фіксація того, які з них відкидаються, для індуктивного відтворення прихованих правил Firewall (ACL), налаштованих на цьому вузлі. На основі симптому періодичного відключення клієнтів від Wi-Fi висунення гіпотези про конфлікт IP-адрес через невірні налаштування DHCP. Перевірка гіпотези шляхом ізоляції сегментів мережі та аналізу ARP-таблиць. Збір та аналіз великого масиву мережевої статистики (логів NetFlow або Wireshark) за добу для індуктивної побудови базової моделі нормального трафіку підприємства (baseline), від якої в подальшому будуть відштовхуватися системи виявлення вторгнень (IDS).
Формалізація відображення об'єктів, явищ або процесів у знаково-символічній формі – за допомогою математичних формул, логічних виразів, мов програмування чи спеціалізованих мов опису апаратури	
ОК Комп'ютерні мережі (3-4 курс)	Переведення текстових вимог замовника («потрібна мережа для трьох відділів на 50, 20 та 12 комп'ютерів») у строгий математичний формат IP-адресації з використанням безкласової маршрутизації (CIDR) та масок змінної довжини (VLSM). Перетворення абстрактної логічної топології мережі (графічної схеми) у строгий, синтаксично безпомилковий скрипт конфігурації (набір команд CLI) для операційної системи маршрутизатора Cisco IOS. Формалізація текстових політик безпеки («заборонити всім доступ до вебсервера, окрім адміністратора») у строгий синтаксис списків контролю доступу (Extended ACL) на рівні портів та протоколів обробки пакетів. Формалізація вимог до якості обслуговування (QoS) для передачі потокового відео у вигляді системи математичних нерівностей (Затримка ≤ 150 ms, Джитер ≤ 30 ms, Втрата пакетів $\leq 1\%$) для подальшого розрахунку необхідної ширини каналу.
Моделювання вивчення об'єкта чи явища шляхом створення та дослідження його копії (моделі), яка заміщує оригінал і відображає його найважливіші властивості	
ОК Комп'ютерні мережі (3-4 курс)	Побудова повної віртуальної копії корпоративної мережі у симуляторі (Cisco Packet Tracer, GNS3) для безпечного тестування роботи протоколів маршрутизації та політик безпеки перед закупівлею і налаштуванням реального дороговартісного обладнання. Робота з готовою імітаційною моделлю топології мережі: штучне зниження пропускну здатності (Bandwidth) головного каналу або відключення кореневого комутатора (Root Bridge) для прогнозування того, де виникне «вузьке місце» (bottleneck) при переспрямуванні трафіку резервними шляхами.

У Таблиці 2.3 представлено приклади завдань для формування навичок використання загальнонаукових методів пізнання в контексті двох освітніх компонентів. Комплекси подібних завдань для інших освітніх компонентів фахової підготовки майбутніх фахівців з комп'ютерної інженерії надано у Додатку В.4.1.

Важливо підкреслити, що протягом усіх років навчання у ЗВО здобувачі вищої освіти опановують специфічні методи реалізації завдань професійної діяльності у галузі комп'ютерної інженерії. Вже на початковому етапі навчання майбутні фахівці з комп'ютерної інженерії мають отримати уявлення про варіативність використання різних методів для вирішення практичних завдань у межах відповідної професійної галузі. Так, наприклад, у контексті теми «Структури повторення в мові С:» (ОК «Програмування», 1 курс) акцентуємо увагу на тому, що для вирішення проблеми оптимізації використання оперативної пам'яті та підвищення швидкодії коду можуть бути використані методи – статичного й динамічного аналізу вихідного тексту, рефакторингу коду, а також методи покрокового програмного налагодження й дебагінгу. Відповідно, актуальним є формування у першокурсників первинних навичок декомпозиції складних інженерних задач та застосування інструментальних засобів автоматизованого пошуку помилок у програмних модулях ще до моменту вивчення складних системних архітектур.

У підрозділі 2.1 ми наголошували на принципі концентричного ускладнення введення таких методів у професійну підготовку фахівців зазначено-го профілю (Рис. 1.2). Підкреслимо ще раз, що володіння загальнонауковими методами є підґрунтям для опанування такими специфічними методами комп'ютерної інженерії, як: системного та архітектурного моделювання, схемотехнічної симуляції, автоматизованого проєктування та конструювання цифрових пристроїв (CAD/EDA); апаратно-програмного копроєктування для розробки вбудованих систем; мережевого планування, налаштування протоколів маршрутизації та віртуалізації інфраструктури (зокрема проєктування хмарних і розподілених архітектур); системного

програмування, паралельних обчислень та інтелектуалізації обчислювальних процесів; вимірювання, технічної діагностики й комплексної верифікації (формальної верифікації, апаратного тестування та налагодження, імітаційного й стрес-тестування, оцінки параметрів надійності); забезпечення кібербезпеки, криптографічного захисту інформації та побудови відмовостійких систем.

Важливим аспектом реалізації завдань формувального етапу експериментальної роботи є включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи. Розроблено у впроваджено в освітній процес підготовки здобувачів спеціальності 123 «Комп'ютерна інженерія» **пакети кейсів різного типу** (аналітичні, техніко-діагностичні, проєктно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), **завдань щодо проблематизації інженерної дійсності**.

Науково-дослідні та аналітичні кейси – це навчальні ситуації, спрямовані на формування методологічної компетентності шляхом моделювання діяльності експерта-аналітика. У цих кейсах здобувач виступає не як виконавець алгоритму, а як суб'єкт, що оцінює архітектурні рішення, виявляє приховані технологічні ризики та обґрунтовує вибір оптимального стеку. Основна мета навчити майбутнього інженера працювати з неоднозначними даними, критично інтерпретувати технічну документацію та синтезувати власне бачення на основі наукових підходів. Замість створення продукту, здобувач формує методологічне обґрунтування (чому саме цей стек? чому саме ця архітектура?), що є критичним елементом професійної зрілості. Найкращий момент для використання таких кейсів – етап закріплення знань після опанування базової теорії дисципліни, тому що здобувач освіти вже володіє понятійним апаратом (термінологією, класифікацією, принципами), тому тепер може проводити порівняльний аналіз (наприклад, DRAM vs SRAM) на рівні аргументованої експертизи, а не простого переказу теорії.

Так, у межах освітнього компоненту «Архітектура комп'ютерів» пропоновано кейс **«Аналіз та вибір підсистеми пам'яті для автономного агро-дрона (Edge AI)»**. Контекст (сюжет) – «Інженерна компанія розробляє нову модель автономного безпілота для точного землеробства. Дрон обладнано нейромережовим прискорювачем для розпізнавання бур'янів у режимі реального часу (Edge Computing)». Відповідно до теоретико-методичних засад використання кейс-технологій студентам була надана проблемна ситуація, інформаційна база та завдання. Проблема ситуація. На етапі проектування апаратної платформи виник конфлікт. Відділ розробки алгоритмів вимагає максимальної пропускної здатності пам'яті для швидкого оброблення зображень. Відділ апаратного забезпечення наголошує на жорстких обмеженнях енергоспоживання, оскільки збільшення ваги акумулятора зменшує час польоту. Необхідно обрати між використанням виключно вбудованої SRAM-пам'яті (дорого, але енергоефективно і швидко) або комбінації невеликого кешу SRAM із зовнішньою LPDDR4-пам'яттю (дешевше, більший обсяг, але вищі затримки та енергоспоживання). Інформаційна база. Технічні специфікації процесорних ядер ARM Cortex-A та NPU. Графіки енергоспоживання LPDDR4 у режимах читання/запису та очікування. Характеристики пропускної здатності (Bandwidth) та затримок (Latency) для обох типів пам'яті. Профіль використання пам'яті (Memory profiling) цільової нейромережі. Завдання: 1) проаналізувати надані графіки енергоспоживання та профілі доступу до пам'яті; 2) розрахувати теоретичний енергетичний бюджет для обох архітектурних рішень під час одногодинного польоту; 3) сформулювати експертний висновок з обґрунтуванням оптимальної архітектури пам'яті для цього специфічного вбудованого пристрою.

Техніко-діагностичні кейси – це навчальні моделі, що відтворюють критичні ситуації системних збоїв та відмов у функціонуванні програмно-апаратних комплексів. Основна методологічна функція таких кейсів полягає у зануренні здобувача в реальне середовище професійної діяльності, де наявні лише «симптоми» проблеми (логічні записи, дампи пам'яті, критичні показники датчиків напруги чи температури), а не готовий алгоритм вирішення.

У межах ОК «Комп'ютерні мережі» студенти працювали над кейсом **«Критична деградація L2-сегмента: Діагностика широкомовного шторму (Broadcast Storm) та відмови STP»**. Контекст (сюжет): «Ви – мережевий інженер центрального кампусу університету. Інфраструктура рівня доступу побудована на комутаторах Cisco Catalyst 2960. У понеділок о 09:00, з початком робочого дня, користувачі бухгалтерії (VLAN 20) та навчального відділу (VLAN 30) масово скаржаться на відсутність доступу до Інтернету та внутрішньої ERP-системи». Проблемна ситуація – спостерігається сильна деградація продуктивності. Пінги до шлюзу за замовчуванням (Core Switch) втрачаються на 60%, а успішні пакети мають затримку від 2 мс до 3000 мс. Світлодіодні індикатори портів на комутаторах рівня доступу безперервно і синхронно блимають на надзвичайно високій частоті.

Інформаційна база (надається здобувачам як додаток):

Додаток 1 (Фрагмент Syslog з Core Switch):

09:02:15: %SW_MAT-4-MACFLAP_NOTIF: Host 001a.2b3c.4d5e in vlan 20 is flapping between port Gi0/1 and port Gi0/2

09:02:18: %SW_MAT-4-MACFLAP_NOTIF: Host f4aa.bbcc.ddee in vlan 30 is flapping between port Gi0/1 and port Gi0/2

09:05:00: %SYS-3-CPUHOG: Task is running for (2000)msecs, more than (2000)msecs (1/0),process = Spanning Tree.

Додаток 2 (Результат команди show processes cpu sorted на Core Switch):

CPU utilization for five seconds: 99%/15%; one minute: 98%; five minutes: 95%

PID Runtime(ms) Invoked uSecs 5Sec 1Min 5Min TTY Process

120 450030 2304000 195 65.11% 60.02% 58.00% 0 Spanning Tree

Додаток 3 (Топологія). Схема, що показує резервовані лінки між комутаторами Access 1, Access 2 та Core.

Завдання для здобувача: 1) на основі наданих логів ідентифікувати першопричину збою (root cause). Пояснити фізичну та логічну природу процесу, що викликав завантаження CPU на 99%; 2) розробити покроковий алгоритм

траблшутингу (Troubleshooting) для швидкої локалізації та ізоляції проблеми в умовах, коли віддалений доступ (SSH) до комутаторів працює з великими перебоями; 3) сформувати конфігураційні рішення (CLI команди) для превентивного захисту топології Spanning Tree Protocol (BPDU Guard, Root Guard, Storm Control), щоб унеможливити подібні аварії, якщо хтось зі мешканців у гуртожитку або співробітників самовільно з'єднає два розетки одним патч-кордом.

Приклади кейсів інших типів представлено у Додатку В.5.

Запропонована система кейсів створює цілісну вертикаль, яка дозволяє забезпечити поетапне формування фахових компетенцій майбутнього фахівця з комп'ютерної інженерії. Інтеграція різнопланових кейсів – від фундаментального науково-дослідного аналізу та технічної діагностики до проєктного конструювання, стратегічного планування та оперативного реагування – дозволяє здобувачу пройти поетапний розвиток від пасивного засвоєння теоретичних знань до формування професійної суб'єктності. Кожен із розглянутих видів кейсів виконує свою специфічну функцію. Якщо діагностичні завдання розвивають аналітичну точність, то проєктні кейси виховують системне бачення, а ситуаційно-стратегічні та міждисциплінарні сценарії вчать розуміти інженерію як частину складного соціотехнічного та бізнес-середовища. У цілому використання кейс-технології сприяє формуванню праксеологічно-технологічного компонента методологічної компетентності, раціональності та доцільності вирішення професійних ситуацій в умовах невизначеності, здатності системно організовувати робочий процес.

Наступним важливим складником експериментальної роботи було визначено розробку та впровадження **завдань щодо проблематизації комп'ютерної інженерної дійсності**. Перед залученням студентів до виконання таких завдань було проведено підготовчу роботу, яка включала:

по-перше, актуалізацію знань щодо змісту поняття «проблема», сутнісних характеристик проблем різного виду (повсякденні, навчальні, наукові, професійні);

по-друге, представлення проблематизації як «визначення, фіксації базових викликів (перешкод, точок розриву, бар'єрів), подолання яких сприятиме розвитку особистості та професійної діяльності людини в цілому» [10, с. 23];

по-третє, огляд таких *методів проблематизації*, як метод емпатії «Подивіться на світ чужими очима», ресурсний розворот, рефреймінг проблем, «диверсійний аналіз» [10], а також таких методів, як метод 5 «Чому», зміна масштабу;

по-четверте, обговорення *етапів проблематизації* в контексті професійної діяльності фахівців з комп'ютерної інженерії (виявлення розриву або суперечності; деконструкція очевидного (критична перевірка базових припущень); локалізація та формулювання проблеми; визначення причинно-наслідкових зв'язків та аналіз контексту; формулювання гіпотези; визначення напрямів діяльності, формулювання технічного завдання.

У зазначеному контексті пропонуємо студентам звернутися до статті Бу Акар Хіби «Якщо ви не проблематизуєте це, ви не побачите і не зрозумієте» [191], що містить переконливі позиції щодо сутності проблематизації у різних сферах життєдіяльності людини.

У процесі підготовки студентів до виконання завдань щодо проблематизації важливо наголосити на суб'єктивному чиннику сприйняття дійсності, який зумовлено обсягом наявних знань, епістемічними установками особистості, готовністю до сприйняття нового, толерантністю до невизначеності. Унаочнення такого суб'єктивного чинника представлено на Рис. 2.2.

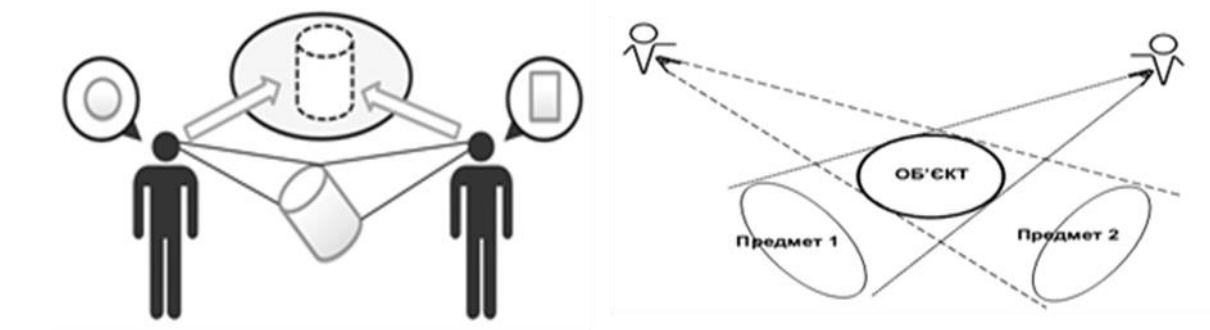


Рис. 2.2. Суб'єктивний чинник сприйняття дійсності [11, с. 580]

Типологія завдань щодо проблематизації комп'ютерної інженерної дійсності будується навколо виявлення та структурування прихованих суперечностей, технологічних обмежень і дефіцитів у тривалій взаємодії «апаратне забезпечення – програмний код – користувач/середовище». Оскільки інженерна дійсність є комплексною, завдання з проблематизації класифікують за об'єктом аналізу, типом системного конфлікту та рівнем абстракції. У Додатку В.6 наведено приклади завдань щодо проблематизації таких *типів*: архітектурно-структурні, ресурсно-технологічні, інтеграційної та еволюційної сумісності, інженерної безпеки та стійкості.

Використання завдань щодо проблематизації сприяє переходу від механічного програмування та налаштування обладнання до глибокого системного мислення. Зіткнення з реальними інженерними компромісами спростовує хибні уявлення здобувачів про існування «ідеальних» технічних рішень. Здобувачі освіти на власному досвіді переконуються, що інженерія – це завжди балансування між суперечливими вимогами: швидкістю та технологічним боргом, безпекою та зручністю, алгоритмічною складністю та апаратними лімітами. Проблема дійсності змушує їх відмовитися від сліпого виконання лабораторних інструкцій на користь аргументованого ухвалення рішень в умовах невизначеності.

Архітектурно-структурні та ресурсно-технологічні завдання демонструють, що просте нарощування потужностей або дублювання ліній зв'язку не рятує від логічних конфліктів чи математичних лімітів алгоритмів, змушуючи майбутніх фахівців покладатися на оптимізацію архітектури, а не на «всемогутнє залізо». Своєю чергою, завдання з інтеграційної сумісності та інженерної безпеки докорінно змінюють парадигму сприйняття мереж. Студенти усвідомлюють складність еволюційних переходів легасі-систем та втрачають ілюзію «безпечного периметра», розуміючи, що справжні загрози часто мімікрують під легальний трафік, який обходить статичні заборони.

Отже, впровадження системи кейсів різного, завдань щодо проблематизації інженерної дійсності в освітній процес є фундаментальною умовою подолання

розриву між академічною підготовкою та реальними вимогами індустрії, адже саме вміння синтезувати технічні, економічні та організаційні аспекти у процесі прийняття рішень є ознакою професійної зрілості сучасного фахівця. Така методологічна гнучкість, сформована через вирішення різнорівневих професійних задач, стає для здобувача методологічним орієнтиром, що забезпечує його адаптивність та ефективність незалежно від подальших технологічних змін.

Важливим складником експериментальної роботи став розроблений та проведений позааудиторний захід – **хакатон «Зворотне проєктування»**, зорієнтований на занурення майбутніх фахівців з комп'ютерної інженерії в умови реальних професійних викликів, формування здатності до проблемного бачення, навичок командної роботи (Додаток В.7).

Підготовчий етап заходу передбачав проєктування сценарію за принципами проблемно зорієнтованого навчання, де ядром виступала сконструйована інженерна суперечність у функціонуванні комп'ютерної системи. З огляду на обмеженість доступу до фізичного апаратного забезпечення в умовах сучасного освітнього процесу, було розроблено комплексне віртуалізоване середовище в симуляторах мережевої інфраструктури (Cisco Packet Tracer / GNS3) або сформовано супровідні артефакти – файли-дампи мережевого трафіку (PCAP) для їх подальшої деконструкції в утиліті Wireshark. Сценарій хакатону свідомо виключав наявність покрокових інструкцій чи готових схем топології, що моделювало ситуацію високої невизначеності та спонукало здобувачів до самостійної пізнавально-дослідницької діяльності.

Безпосереднє проведення хакатону відбувалося за чітко структурованим циклом та передбачало командну взаємодію здобувачів із розподілом кваліфікаційних ролей (наприклад: мережевий аналітик, архітектор, інженер із безпеки). На першому етапі команди здійснювали «розвідку наосліп», висуваючи методологічні гіпотези щодо архітектури заблокованої системи та причин її дестабілізації (зокрема, виявлення широкомовних штормів чи аномалій маршрутизації через аналіз службових пакетів). Роль модератора заходу зводилася до фасилітації: прямі технічні

підказки замінювалися коучинговими запитаннями, що спонукали студентів до рефлексії над власними діями. Перехід до синтезу рішень відбувався лише після успішного захисту відтвореної логічної схеми «чорного ящика», після чого команди отримували доступ до конфігурування обладнання для усунення дефектів та проведення архітектурної оптимізації мережі.

Аналіз результатів проведення хакатону та зворотного зв'язку продемонстрував загалом позитивну реакцію студентської спільноти на таку інноваційну форму організації практичної підготовки. Попри ситуацію пізнавальної невизначеності, викликану відсутністю звичного лабораторного алгоритму та прямих інструкцій викладача, здобувачі виявили високу мотивацію, спортивний азарт та залученість до процесу інженерного пошуку. Робота в команді дозволила їм компенсувати дефіцит індивідуальних знань за рахунок колективного обговорення та спільної верифікації гіпотез.

У процесі підсумкового інтерв'ювання студенти зазначали, що формат зворотного проєктування дозволив їм подолати шаблони репродуктивного мислення та наочно побачити причинно-наслідкові зв'язки між абстрактними протоколами й апаратною поведінкою систем. Проведення заходу засвідчило, що зміщення акцентів із механічного виконання команд на системний аналіз ефективно стимулює формування дослідницького, аналітичного та проєктно-синтетичного компонентів методологічної компетентності майбутніх комп'ютерних інженерів, що повністю підтверджує доцільність упровадження експериментальної методики.

Важливим елементом проблемно зорієнтованого навчання у технічних ЗВО є **залучення студентів до проєктної діяльності**. Нами було розроблено та впроваджено в освітній процес комплекс взаємопов'язаних навчально-дослідницьких проєктів, який охоплює як індивідуальні, так і групові форми навчальної діяльності, забезпечуючи поступовий перехід здобувачів освіти від репродуктивного виконання завдань до самостійного інженерного пошуку. Зокрема, у межах обов'язкової компоненти «Комп'ютерні мережі» реалізовано проєкти, спрямовані на проєктування та комплексне моделювання корпоративної

інфраструктури. Індивідуальний формат вимагає від студента самостійної декомпозиції невизначеного технічного завдання підприємства, проведення математичних розрахунків адресного простору та обґрунтування вибору архітектурних рішень. Логічним продовженням виступає груповий проєкт, що моделює розробку розподіленої відмовостійкої мережі мультифіліального підприємства. У цьому форматі студенти розподіляють між собою функціональні ролі (мережевий архітектор, інженер з безпеки, фахівець з автоматизації), що створює умови для узгодження технічних компромісів, розвитку комунікативних навичок та спільного вирішення системних конфліктів. У Додатку В.10 представлено приклади проєктів, етапи реалізації та критерії оцінювання.

Для розширення інструментарію майбутніх інженерів та інтеграції в їхню діяльність сучасних технологій штучного інтелекту, проєктну базу було доповнено завданнями з вибіркової дисципліни «Основи промпт-інжинірингу». Індивідуальний вимір цієї роботи фокусується на розробці мультикрокових промпт-систем, де студент шляхом ітеративних експериментів та траблшутингу вчиться керувати поведінкою великих мовних моделей для вирішення прикладних фахових задач.

Показові результати у контексті формування методологічної компетентності продемонстрував **командний проєкт з розробки комплексного AI-асистента**, архітектоніка якого повністю спиралася на імплементацію гнучких методологій (Agile) розробки програмного забезпечення (Додаток В.10). Як слушно зазначають І. Гарко та співавтори у своєму дослідженні впровадження Agile-підходів у вищій школі, «модель послідовного викладання матеріалу вже віджила своє в епоху доступу до інформації, і завданням викладачів стає не примітивна вчитка матеріалу, а формування уявлення про гнучкість процесів, що у світлі розробки програмного забезпечення називають Agile-принципами» [22, с. 149]. Спираючись на ці ідеї та досвід адаптації технології Agile до навчального процесу, ми структурували виконання даного проєкту не як каскадне виконання набору вимог, а як серію ітеративних спринтів із використанням фреймворку Scrum (Рис. 2.3).

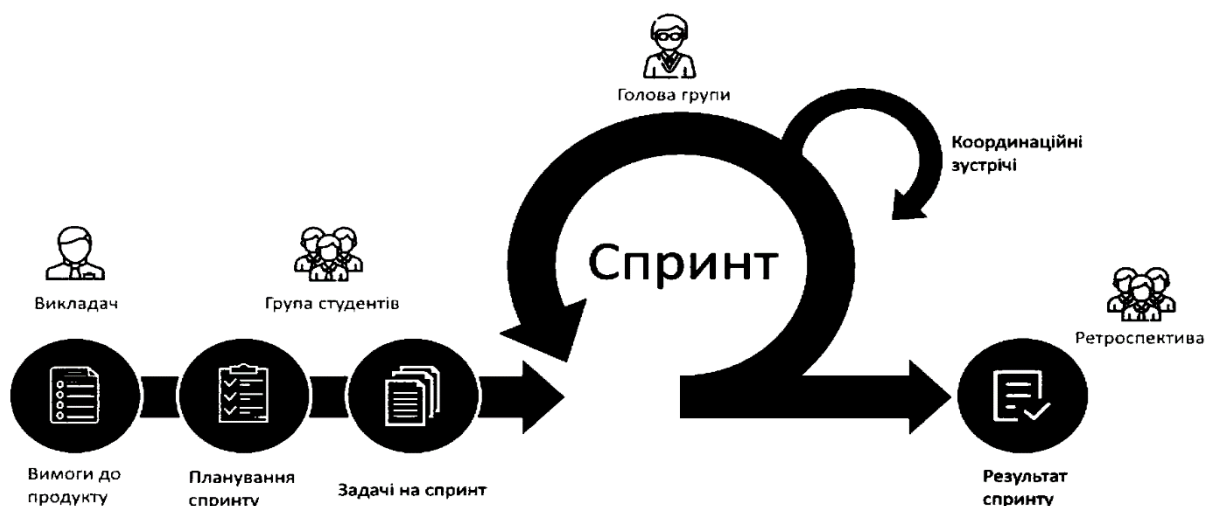


Рис. 2.3. Класична схема eduScrum [22, с. 152]

Здобувачі освіти брали на себе ролі Product Owner, Scrum Master та інженерів із якості, самостійно формуючи беклог продукту та проводячи ретроспективи. Такий підхід довів свою високу педагогічну ефективність: перетворення викладача на консультанта-координатора та імітація реального циклу розробки ІТ-продукту змушували студентів системно аналізувати помилки, нести колективну відповідальність за кінцевий результат та оперативна реагувати на зміни вимог. Загалом, результати спостережень підтверджують, що групова проєктна діяльність на засадах Agile-методології найкраще каталізує розвиток критичного мислення, здатності до фахової рефлексії та вміння працювати в умовах невизначеності, що є фундаментальними характеристиками методологічної компетентності сучасного інженера.

Упровадження четвертої педагогічної умови передбачало розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності. Формування методологічної компетентності виключає пасивне накопичення теоретичної інформації, оскільки цей процес стає можливим лише за умови трансформації студента в активного суб'єкта освітньої діяльності. Завдяки цілеспрямованій актуалізації рефлексивної позиції відбувається глибоке засвоєння зовнішнього освітнього контенту — від фундаментальних теорій до конкретних методів і підходів — та його перетворення

на стійкі внутрішні професійні орієнтири. Саме розвинена рефлексія забезпечує усвідомлення майбутнім спеціалістом власної суб'єктності, що дозволяє йому автономно діагностувати дефіцит конкретних знань, оперативно моделювати й виправляти індивідуальний освітній маршрут, а також цілком усвідомлено контролювати процес нарощування професіоналізму на всіх етапах підготовки.

Важливо підкреслити, що орієнтація на формування рефлексивної позиції сучасних студентів – це наскрізне завдання закладу вищої освіти, що має бути актуальним на будь-якому етапі навчання, у процесі використання усіх форм та методів організації освітньої діяльності.

Відповідно до завдань дослідження у практиці аудиторної та позааудиторної роботи із студентами спеціальності 123 «Комп'ютерна інженерія» використовувалися такі форми та методи, що суттєво сприяли становленню та розвитку рефлексивності здобувачів, їхньої рефлексивної позиції у навчальній та науково-дослідницькій діяльності: метод портфоліо (накопичення та публічний аналіз власних програмних продуктів, схемотехнічних рішень та архітектурних проєктів тощо); організація рефлексивної діяльності у процесі вирішення кейсів різного типу; взаєморецензування (практика перевірки коду чи схем один одного (в ІТ-сфері – *Code Review*)); обговорення проміжних результатів діяльності у межах командної роботи (міні-хакатони, воркшопи).

Особливу увагу у процесі практичної роботи із студентами було приділено веденню **е-портфоліо** (Додаток В.9). Традиційний підхід, за якого оцінюється лише фінальний програмний код або конфігураційний файл, залишає поза увагою сам когнітивний процес пошуку рішення під час проєктування масштабованої мережевої топології чи оптимізації бази даних. Натомість портфоліо дозволяє системно зафіксувати хід дослідницького пошуку: проаналізувати альтернативи, аргументувати причини відхилення певних алгоритмів чи протоколів та відобразити проміжні результати.

Водночас, в умовах використання технологій проблемно зорієнтованого навчання, портфоліо фактично трансформується у своєрідний «методологічний щоденник», який забезпечує нерозривний зв'язок між фундаментальною теорією

та апаратною дійсністю. Оскільки в процесі вирішення завдань студенти неминуче стикаються з реальними інженерними суперечностями, вони отримують простір для документування власного досвіду перенесення ідеалізованих теоретичних моделей на практичні, часто неідеальні об'єкти. Це особливо важливо у ситуаціях, коли виникає пізнавальна напруга між очікуваними метриками – наприклад, коли показники систем моніторингу розходяться з реальною пропускну здатністю обладнання.

Портфоліо має наскрізну модульну структуру. Це означає, що воно наповнюється поетапно в межах вивчення окремих освітніх компонентів, де за кожен сформований інженерний кейс здобувач отримує поточну оцінку. На завершальних етапах навчання розрізнені модулі інтегруються в єдину, цілісну екосистему, яка демонструє здатність майбутнього інженера вирішувати комплексні міждисциплінарні завдання. Фрагмент портфоліо здобувача представлено у Додатку В.9.

Таким чином, упровадження педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання дозволило включити здобувачів вищої освіти до різних форм квазіпрофесійної діяльності на основі технологій проблемно зорієнтованого навчання, сприяло формуванню методологічних знань умінь та навичок як підґрунтя успішної майбутньої професійної діяльності в ІТ-сфері.

2.3. Аналіз результатів дослідно-експериментальної роботи

Формувальний етап дослідно-експериментальної роботи з формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії здійснювався на підставі теоретико-методичних засад дослідження, визначених педагогічних умов та відповідної критеріально-рівневої бази.

Експериментальна робота ґрунтувалася на комплексі взаємопов'язаних методів: аналіз наукових та навчально-методичних джерел та нормативних

документів, систематизація та узагальнення наукових положень для обґрунтування відповідних педагогічних умов, анкетування, тестування, самооцінка, кейс-тестінг для визначення рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії; педагогічний експеримент для доведення ефективності пропонованих умов; методи математичної статистики для визначення статистичної значущості результатів дослідження.

У підрозділі 1.4. наведено критерії та показники, на основі яких здійснювалося дослідження рівнів сформованості відповідної компетентності: мотиваційний, когнітивний, діяльнісний, рефлексивний та комунікативний. Визначені критерії та показники зумовлюють цілісний підхід до діагностичних процедур.

Мотиваційний критерій відображає сукупність мотиваційних настанов фахівців з комп'ютерної інженерії до здійснення успішної професійної діяльності, забезпечення її методологічних основ та увиразнено такими показниками: розуміння значущості методологічної компетентності для майбутньої професійної діяльності, наявність мотивації на успіх та потреби в досягненні, пізнавальні потреби та інтерес до методологічної діяльності.

Когнітивний критерій представлено такими показниками: знання загальної методології та методології інженерної діяльності, знання щодо загальнонаукових методів пізнання та методів інженерної діяльності.

Діяльнісний критерій відображає праксеологічно-технологічний складник методологічної компетентності та включає такі показники: Уміння збирати інформацію з різних джерел, оцінювати її якість, достовірність, уміння та навички узагальнювати інформацію, здатність до аналогії, класифікації та аналізу, критичність мислення, толерантність до невизначеності.

Рефлексивний критерій пов'язано із умінням критично оцінювати результати своєї діяльності, здатністю до рефлексії, потребами у саморозвитку та самовдосконаленні.

Комунікативний критерій відображає здатність працювати в команді, наявність комунікативних та організаторських здібностей, здатність обирати

адекватні методи, стилі та канали вербальної і письмової комунікації залежно від рівня технічної підготовки аудиторії.

Представлені критерії репрезентують сутнісні характеристики методологічної компетентності майбутніх фахівців з комп'ютерної інженерії та дозволили визначити відповідні її рівні – високий, середній та низький, змістові характеристики яких надано у підрозділі 1.4.

У процесі реалізації діагностичних процедур як на початковому констатувальному, так і на підсумковому контрольному етапах дослідження ключова, увага приділялася відстеженню динаміки змін у рівнях сформованості методологічної компетентності за визначеними критеріями впродовж усього періоду експериментальної роботи. Важливим аспектом стало урахуванням взаємозумовленості та тісного взаємозв'язку між усіма структурними компонентами методологічної компетентності, що зумовило потребу в ретельному аналізі покомпонентної динаміки досліджуваного процесу на базі відповідного критеріального інструментарію. Крім того, логіка дослідження передбачала обов'язкове проведення порівняльного аналізу результатів, отриманих на констатувальному та формуальному етапах експерименту, що відображає традиційну дослідницьку схему в межах педагогічного пошуку. Ефективність цього процесу забезпечувалася застосуванням взаємодоповнювальних методів і методик діагностики. Водночас обов'язковим чинником є єдність якісного та кількісного аналізу накопичених емпіричних даних із урахуванням індивідуальної траєкторії та динаміки розвитку методологічної компетентності кожного студента закладу вищої освіти.

Дослідно-експериментальна робота проводилась упродовж 2023-2026 рр. із здобувачами першого (бакалаврського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології». Національного університету біоресурсів і природокористування України, ДЗ «Луганський національний університет імені Тараса Шевченка», Державного університету «Київський авіаційний інститут».

Відповідно до логіки дослідження в контрольних групах (КГ) навчання здійснювалося за традиційним форматом освітнього процесу, а в експериментальних (ЕГ) – відповідно до розроблених та обґрунтованих педагогічних умов.

Дослідно-експериментальна робота здійснювалась у три етапи – констатувальний, формувальний та контрольний.

На констатувальному етапі здійснено виявлення рівня сформованості методологічної компетентності за визначеними критеріями (мотиваційний, когнітивний, діяльнісний, рефлексивний та комунікативний) в КГ та ЕГ, аналіз освітніх програм, навчальних планів, навчально-методичного забезпечення освітніх компонентів підготовки бакалаврів за спеціальністю 123 «Комп'ютерна інженерія».

Формувальний етап експерименту був спрямований на визначення ефективності розроблених і теоретично обґрунтованих педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії під час опанування ними профільних дисциплін, а також на практичну перевірку висунутої на початку наукового пошуку гіпотези. Протягом цього періоду особливий акцент робився на виявленні найбільш дієвих форм аудиторної та позааудиторної діяльності студентів у процесі вивчення освітніх компонентів загального та фахового блоків освітньої програми, а також на оцінюванні розвивального потенціалу проблемно зорієнтованих завдань, завдань щодо проблематизації інженерної дійсності, кейсів різного типу тощо, орієнтованих на відпрацювання методологічних умінь та навичок. У ході практичного впровадження визначеного комплексу педагогічних умов забезпечувалася систематична фіксація труднощів, які виникали в майбутніх інженерів під час роботи з конкретними типами завдань, розв'язання спеціалізованих міні-кейсів або в процесі проєктної діяльності і технічних рішень різного рівня складності.

На підсумковому етапі дослідження було проведено порівняльно-змістовий аналіз діагностичних вимірювань. Відповідно, контрольний етап

передбачав опрацювання отриманих результатів, визначення їхнього практичного значення.

Результати дослідно-експериментальної роботи уможливили висновок про те, що розроблені педагогічні умови формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованого навчання є ефективними. Порівняння початкового та підсумкового вимірювання засвідчують достовірність такої ефективності. Використання комплексу діагностичних методів та методик дозволило забезпечити об'єктивність та достовірність отриманих даних.

Результати діагностичних зрізів на констатувальному та контрольному етапах дослідження представлено у Таблиці 2.4.

Таблиця 2.4

Динаміка рівні сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії (%)

Групи	Рівень	Критерії									
		Мотиваційний		Когнітивний		Діяльнісний		Рефлексивний		Комуніка- тивний	
		Етап експерименту									
		Конст.	Контр.	Конст.	Контр.	Конст.	Контр.	Конст	Контр	Конст	Контр
ЕГ	В	40,43	57,45	12,77	31,91	17,02	36,17	19,15	38,30	17,02	38,30
	С	42,55	36,17	40,43	53,19	46,81	51,06	48,94	51,06	48,94	51,06
	Н	17,02	6,38	46,81	14,89	36,17	12,77	31,91	10,64	34,04	10,64
КГ	В	37,25	41,18	13,73	17,65	15,69	19,61	17,65	23,53	15,69	19,61
	С	39,22	43,14	43,14	45,10	45,10	47,06	50,98	49,02	49,02	50,98
	Н	23,53	15,69	43,14	37,25	39,22	33,33	31,37	27,45	35,29	29,41

Примітка. В – високий рівень, С – середній рівень. Н – низький рівень

У таблиці унаочнено результати підсумків констатувального та підсумкового етапів дослідно-експериментальної роботи за кожним із критеріїв та групами (КГ, ЕГ) та відповідними рівнями (високий, середній та низький) сформованості методологічної компетентності майбутніх фахівців з

комп'ютерної інженерії. За результатами констатувального зрізу отримано дані про майже ідентичність рівнів у контрольній та експериментальній групах.

Узагальнення результатів свідчить про суттєві зміни у показниках за відповідними критеріями в студентів ЕГ.

Для визначення показників за мотиваційним критерієм використовувалися запитання Анкети самооцінки, методика «Мотивація до успіху» Т. Елерса, тест-опитувальник «Потреба у досягненні». Так, кількість студентів з високим рівнем мотивації до успіху в ЕГ зросла до 55,3% (проти 36,17% на констатувальному етапі), у той час як показник середнього рівня склав 42,5%, а низького – зменшився до мінімальних 2,1%. У КГ зміни виявилися менш суттєвими: високий рівень діагностовано у 39,2% здобувачів, середній – у 50,9%, а низький рівень – у 9,8% осіб.

Аналіз результатів за тестом-опитувальником «Потреба у досягненні» на контрольному етапі засвідчив суттєві позитивні зміни в ЕГ. Кількість респондентів ЕГ із високим рівнем потреби у досягненні зросла до 61,70% (проти 44,7% на констатувальному етапі). Показник середнього рівня склав 27,7%, а кількість студентів із низьким рівнем зменшилася більше ніж удвічі – до 10,64% (проти 21,3% на початку експерименту). У КГ спостерігається значно слабша динаміка: високий рівень діагностовано у 45,10% досліджуваних, середній – у 31,37%, а низький рівень залишився на високій позначці й склав 23,53%. Отримані кількісні дані підтверджують дієвість педагогічних умов, упроваджених у процесі підготовки студентів ЕГ.

У цілому за мотиваційним критерієм кількість студентів із високим рівнем збільшилась на 17,02%, натомість у КГ приріст – 3,92%. Кількість студентів ЕГ із середнім рівнем зменшилась на 6,38%, відповідно із низьким зменшилась на 10,65%. Показники за середнім та низьким рівнем в студентів КГ такі: збільшення за середнім рівнем – 3,92%, зменшення за низьким – 7,84%. Така динаміка в ЕГ зумовлена цілеспрямованою роботою щодо стимулювання інтересу до методологічних проблем інженерної діяльності, залученням студентів до різних форм аудиторної та позааудиторної роботи

щодо висвітлення перспектив успішної діяльності в ІТ-сфері. Незначна динаміка в КГ зумовлена природним дорослішанням та подальшою адаптацією студентів до особливостей освітнього процесу у ЗВО.

Відповідно до сутнісних ознак когнітивного компонента методологічної компетентності майбутніх фахівців з комп'ютерної інженерії важливим показником було визначено адекватне розуміння таких понять, як «методологія», «метод», «методологія комп'ютерної інженерії», «проблема», а також сутності та умов використання специфічних методів професійної діяльності фахівців у галузі комп'ютерної інженерії. На контрольному етапі дослідження використовувалися позиції Анкети самооцінки, результати виконання поточних підсумкових тестів та лабораторних робіт за освітніми компонентами навчального плану. Особливу увагу було приділено точності та доцільності використання технічної термінології, обґрунтуванню студентами доцільності використання загально-наукових та специфічних методів у межах виконання проблемних завдань, аналізу кейсів різного типу, індивідуальних та групових проєктів й т.п.

За результатами відповідей на запитання анкети «Як ви розумієте значення поняття «методологія» у контексті вашої майбутньої професії комп'ютерного інженера?» у ЕГ частка студентів із високим рівнем (усвідомлення методології як системи принципів і методів, що визначає стратегію інженерного мислення та дозволяє свідомо обирати найкращі підходи й інструменти розробки) зросла з 10,6% на констатувальному етапі до 27,66% на контрольному. Кількість здобувачів із середнім рівнем (розуміння методології як готової інструкції, алгоритму чи стандарту виконання проєкту) збільшилася з 27,6% до 40,4%. Водночас частка студентів із низьким рівнем (сприйняття методології як суто академічного формального терміну, відірваного від реальної інженерної практики) скоротилася майже вдвічі – з 61,7% до 31,91%. У КГ позитивні зміни виявилися значно слабшими: кількість студентів з високим рівнем розуміння методології зросла лише з 9,8% до 13,72%, із середнім рівнем – з 27,45% до 29,41%, а низький рівень зменшившись із 62,75% лише до 56,86%.

У процесі методологічної підготовки здобувачів вищої освіти важливим є адекватне розуміння сутності загальнонаукових методів пізнання. Так, на запитання «Що для вас означає виконати аналіз певного об'єкта, явища чи процесу?» у ЕГ частка студентів із високим рівнем (розуміння аналізу як розкладання цілого на елементи з обов'язковим дослідженням внутрішніх причинно-наслідкових зв'язків та виявленням головного чинника) зросла з 23,4% на констатувальному етапі до 34,04% на контрольному. Кількість здобувачів із середнім рівнем (сприйняття аналізу як розчленування об'єкта на частини для детального вивчення) збільшилася з 57,45% до 65,96%. Водночас низький рівень (поверхнєве ознайомлення з інформацією та її переказ) в експериментальній групі не було виявлено. У КГ позитивні зміни виявилися мінімальними: частка студентів з високим рівнем розуміння аналізу зросла з 21,57% до 25,49%, із середнім рівнем – з 58,82% до 60,78%, а низький рівень скоротився з 19,61% лише до 11,76%».

Так само, достатньо впевнено студенти ЕГ надали відповідь на запитання щодо сутності синтезу як наукового методу пізнання. В ЕГ частка студентів із високим рівнем (розуміння синтезу як поєднання елементів у нове цілісне утворення для отримання якісно нового знання чи створення нового об'єкта) зросла з 23,40% на констатувальному етапі до 29,79% на контрольному. Кількість здобувачів із середнім рівнем (сприйняття синтезу як об'єднання розрізнених частин чи понять) збільшилася з 63,83% до 68,09%. Водночас низький рівень (сприйняття синтезу як механічного копіювання існуючих рішень за зразком) в ЕГ було повністю ліквідовано – показник зменшився з 12,77% до 0,00%. У КГ динаміка виявилася незначною: частка студентів з високим рівнем розуміння синтезу зросла з 23,53% до 27,45%, із середнім рівнем – з 62,75% до 66,67%, а низький рівень скоротився з 13,73% до 5,88%».

Оскільки запитання анкети «Що таке проблема? Чи є відмінності у розумінні поняття «проблема» на побутовому, навчальному, науковому та професійному рівнях?» було відкритим, то відповіді студентів групувалися за наближенням до словникових визначень. Типова відповідь високого рівня –

«Проблема – це не просто труднощі чи перешкода, а фундаментальна суперечність між наявним станом системи (або знань про неї) та необхідним (бажаним) результатом, для розв'язання якої в суб'єкта наразі відсутній готовий алгоритм чи достатній обсяг інформації. З методологічного погляду проблема виникає тоді, коли відомі методи не дозволяють досягти мети, що стимулює пошук нових принципів, моделей або інструментів. Побутовий рівень: Проблема сприймається суб'єктивно – як емоційний або практичний дискомфорт, дилема чи конкретна життєва перешкода (наприклад, зламався ноутбук або зник зв'язок). Навчальний рівень – проблема виступає як дидактичний інструмент. Це навчально-пізнавальна задача з контрольованим рівнем невизначеності (наприклад, кейс із оптимізації алгоритму). На науковому рівні проблема має статус «знання про незнання», що об'єктивно відображає наявну суперечність у теорії чи фундаментальних законах функціонування систем, яку неможливо усунути в межах існуючої наукової парадигми. На професійному рівні проблема формулюється як комплексний техніко-технологічний виклик або системний баг. Це суперечність між вимогами до продукту (продуктивність, енергоефективність, безпека) та наявними апаратними чи програмними обмеженнями (наприклад, забезпечення нульової затримки передачі даних у критичних IoT-системах)». У цілому було отримано такий розподіл. У ЕГ частка студентів, які продемонстрували високий рівень (надали системне визначення проблеми як суперечності та чітко диференціювали її характеристики на всіх чотирьох рівнях із наведенням фахових прикладів з комп'ютерної інженерії), зросла з 10,64% на констатувальному етапі до 27,6% на контрольному етапі. Показник середнього рівня (повноцінне розуміння сутності проблеми з частковим розрізненням рівнів) збільшився з 42,55% до 55,3%. Разом з тим, кількість здобувачів із низьким рівнем (побутовою трактовкою проблеми як звичайної перешкоди без усвідомлення її контекстуальних рівнів) скоротилася майже втричі – з 46,8% до 17,02%. У КГ позитивні зміни виявилися значно слабшими: кількість студентів з високим рівнем зросла з 9,8% до 13,73%, із

середнім рівнем – з 43,1% до 45,10%, а низький рівень залишився переважаючим, зменшившись із 47,1% лише до 41,18%».

Обсяг підрозділу не дозволяє навести кількісні показники за усіма запитаннями анкети. Зазначимо, що кількісна динаміка у цілому є типовою порівняно з наведеними прикладами.

Порівняння результатів сформованості методологічної компетентності за когнітивним критерієм в ЕГ та КГ наприкінці експериментальної роботи засвідчує позитивну динаміку на користь студентів ЕГ. Так, зафіксовано збільшення кількості здобувачів із високим рівнем – на 19,15%, за середнім рівнем – на 12,8%, а також суттєве зменшення за низьким рівнем – на 31,9 %. Наведені результати щодо показників за цим критерієм в КГ значної динаміки не висвітлюють. Кількість студентів КГ, що показали високий рівень зростає тільки на 3,92%, середній рівень – на 1,96%. Відзначимо, що кількість студентів із низьким рівнем все ж таки зменшилась на 5,9%.

Підвищення показників за когнітивним критерієм у студентів ЕГ свідчить про опанування знань з проблем загальнонаукових методів пізнання, усвідомленого використання методів інженерної діяльності, володіння основними технологіями пошуку та обробки інформації. При виконанні проблемних завдань, вирішенні кейсів різного типу, проектних завдань як типових форм проблемно зорієнтованого навчання студенти значно більш активно демонстрували усвідомлені знання щодо алгоритмічних приписів (правил) використання методів аналізу, синтезу, класифікації, узагальнення тощо, виявляли зацікавленість динамікою науково-технологічних рішень в ІТ-сфері.

Для діагностики сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за діяльнісним критерієм використовувалися такі методики: кейс-тестінг «Самооцінка умінь та навичок роботи з інформацією» (Додаток Б.3), методики «Дослідження узагальнення як мисленнєвої операції», «Логічно-понятійне мислення. Утворення складних аналогій», «Виключення понять» (здатність до класифікації та аналізу) (Додатки Б.4 – Б.6), методики на визначення рефлексивності та критичності

мислення (Додаток Б.7), опитувальника С. Баднера «Виявлення толерантності до невизначеності» (Додаток Б.8).

Кейс-тестінг «Самооцінка умінь та навичок роботи з інформацією» дозволив отримати узагальнювальні позиції щодо такого показника, як «Уміння збирати інформацію з різних джерел, оцінювати її якість, достовірність». У ЕГ кількість здобувачів з високим рівнем зросла до 34,04% (проти 14,8% на констатувальному етапі), із середнім рівнем – до 44,7% (проти 38,3%). Відповідно, відбулося значне скорочення кількості студентів з низьким рівнем – до 21,3% (проти 46,8% на початку дослідження). У КГ спостерігалася значно менш виражена динаміка: високий рівень виявили 21,57% здобувачів (проти 15,7%), середній – 37,2% (проти 39,2%), а низький рівень зменшився лише до 41,2% (проти 45,1% на констатувальному етапі).

Зіставлення результатів дослідження на контрольному етапі за методикою «Дослідження узагальнення як мисленнєвої операції» засвідчило виражену позитивну динаміку сформованості відповідної мисленнєвої операції в ЕГ у межах діяльнісного критерію. Так, у ЕГ кількість студентів із високим рівнем навичок узагальнення зросла з 19,15% на констатувальному етапі до 40,43% на контрольному етапі, а із середнім рівнем – з 51,06% до 55,32%. Показники низького рівня в ЕГ зазнали суттєвого скорочення – з 29,79% до 4,26%. Натомість у КГ зміни мають незначний характер: частка здобувачів із високим рівнем збільшилася лише з 17,65% до 21,57%, із середнім рівнем – з 50,98% до 52,94%, а низький рівень зменшився з 31,37% до 25,49%.

Порівняльний аналіз результатів за методикою «Логічно-понятійне мислення. Утворення складних аналогій» на контрольному етапі засвідчив такі результати. В ЕГ частка здобувачів з високим рівнем логічно-понятійного мислення зросла з 17,02% на констатувальному етапі до 36,17% на контрольному, а із середнім рівнем – з 44,68% до 51,06%. Водночас показники низького рівня в ЕГ зменшилися більш ніж утричі – з 38,3% до 12,77%. Натомість у КГ динаміка виявилася незначною: частка студентів з

високим рівнем зростає з 17,65% до 19,61%, із середнім рівнем – з 43,14% до 45,1%, а низький рівень зменшився лише з 39,22% до 35,29%.

Методика «Виключення поняття» дозволяє здійснити аналіз здатності студентів розмежовувати суттєві та другорядні ознаки, розуміти родо-видові відносини між поняттями. На контрольному етапі дослідження в ЕГ частка здобувачів, які демонструють високий рівень, зростає з 17,02% на констатувальному етапі до 38,3% на контрольному етапі, а із середнім рівнем – з 48,94% до 53,19%. Водночас показники низького рівня в ЕГ скоротилися майже в чотири рази – з 34,1% до 8,5%. У КГ динаміка виявилася незначною: кількість студентів з високим рівнем зростає з 17,65% до 21,6%, із середнім рівнем – з 47,06% до 49,02%, а низький рівень зменшився з 35,29% до 29,4%.

У контексті нашого дослідження особлива увага була приділена такій якості сучасних фахівців, як толерантність до невизначеності, що пов'язано із особливими умовами технологічного розвитку усіх сфер суспільства в цілому та ІТ-сфери зокрема. Використовувалася достатньо відомий опитувальник С. Баднера «Виявлення толерантності до невизначеності». У ЕГ кількість здобувачів із високим рівнем толерантності до невизначеності зростає з 17,02% на констатувальному етапі до 36,17% на контрольному етапі. Кількість студентів із середнім рівнем збільшилася з 46,8% до 53,2%. Водночас показники низького рівня (сприйняття ситуацій новизни та складності як загрози й джерела стресу) в ЕГ скоротилися більш ніж утричі – з 36,2% до 10,6%. У КГ спостерігалася значно менш виражена динаміка: високий рівень продемонстрували 19,6% здобувачів (проти 15,7%), середній – 47,1% (проти 47,1%), а низький рівень зменшився лише до 33,3% (проти 37,25% на початку дослідження).

У цілому за показниками діяльнісного критерію сформованості методологічної компетентності динаміка відповідних рівнів в студентів ЕГ наступна. На 19,5% збільшилась кількість здобувачів, які виявили високий рівень, на 4,26% – середній рівень. Відповідно, зменшилась кількість студентів із низьким рівнем – на 23,4%. Показники в КГ продемонстрували значно повільнішу динаміку: збільшення кількості студентів із високим

рівнем – на 3,9%, із середнім рівнем – на 1,96%, зменшення кількості студентів із низьким рівнем – тільки на 5,9%. Більш виражена динаміка у студентів ЕГ засвідчила сформовані вміння щодо проблематизації інженерної дійсності, бачення протиріч та аналізу утруднень у різних формах проєктної діяльності, опанування базовими інженерно-методологічними алгоритмами.

Для дослідження рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за рефлексивним критерієм на контрольному етапі використано запитання Анкети самооцінки (Додаток А), методику визначення індивідуальної міри вираженості рефлексивності (А. Карпов) (Додаток Б.9), анкету «Здатність до саморозвитку» (Додаток Б.10), спостереження та бесіди за результатами залучення студентів до проблематизації, аналізу проблемних ситуацій, кейсів різного типу, участі в таких інтерактивних формах аудиторної та позааудиторної роботи, як міні-хакатони, виконання індивідуальних та групових проєктів, ведення е-портфоліо.

На запитання анкети «Наскільки ви внутрішньо готові до того, що через стрімкий розвиток комп'ютерних інженерних технологій вам доведеться постійно переучуватися та освоювати нові методології розробки протягом усієї кар'єри?» кількість студентів ЕГ, які обрали варіант високого рівня («повністю готові, оскільки динаміка змін їх надихає...»), зросла до 38,3% (у КГ – 21,57%). Варіант середнього рівня («Сприймаю це як неминучу необхідність...») обрали 51,1% здобувачів ЕГ та 50,9% КГ. Частка студентів із низьким рівнем готовності («Не готовий, це мене демотивує...») в ЕГ суттєво скоротилася до 8,51%, тоді як у КГ вона залишається високою і становить 27,45%. Аналогічні позитивні зміни спостерігаються і під час аналізу відповідей на інше запитання («Чи аналізуєте ви методи, якими користуєтеся для навчання, пошуку інформації та вирішення інженерних завдань...»). Варіант високого рівня («Періодично аналізую свій робочий процес...») на контрольному етапі обрали 38,3% студентів ЕГ проти 21,57% у КГ. Показники за середнім рівнем («Іноді помічаю, що певні методи неефективні, але рідко намагаюся змінити...») становлять 51,06% в ЕГ та 50,9% у

КГ. Варіант низького рівня («Ні, я дію за звичкою і ніколи не замислююся...») обрали лише 8,5% студентів ЕГ, у той час як у КГ цей показник становить 27,4%.

За результатами використання методики визначення індивідуальної міри вираженості рефлексивності (А. Карпов) в ЕГ частка студентів з високим рівнем рефлексивності, що характеризується стійкою здатністю до глибокого самоаналізу, зросла до 36,2% (приріст склав 21,0%). Середній рівень, який відображає здатність до ситуативної рефлексії, зафіксовано у 48,9% респондентів (збільшення на 0,5%). Водночас кількість студентів із низьким рівнем суттєво зменшилась на 21,5% і становить 14,89%, що свідчить про подолання більшістю здобувачів труднощів у прогнозуванні наслідків власних дій та критичному переосмисленні досвіду. У КГ динаміка виявилася значно менш вираженою: високий рівень рефлексивності виявлено у 19,6% студентів, середній рівень – у 47,06%, а низький рівень зменшився на 3,6% і становить 33,3%.

Важлива роль у становленні рефлексивної позиції майбутніх фахівців з комп'ютерної інженерії належить здатності до саморозвитку, яка дозволяє їм неперервно підвищувати професійний рівень, опановувати нові технології в умовах стрімкого розвитку ІТ-галузі. Методологічна компетентність перетворює цей процес на системну діяльність, забезпечуючи інженера алгоритмами свідомого вибору інструментів, критичного аналізу рішень та ефективного проєктування складних обчислювальних систем. Використання анкети «Здатність до саморозвитку» дозволило отримати такі результати. В ЕГ частка студентів, які досягли високого рівня (стадія «активного розвитку») із чіткою внутрішньою мотивацією до самовдосконалення, зросла на 21,0% і становить 40,43%. На середньому рівні, що характеризується нестабільною орієнтацією на розвиток залежно від зовнішніх стимулів, зафіксовано 53,19% респондентів. Показник низького рівня («зупинка розвитку») зазнав істотного скорочення на 21,5% і становить лише 6,38%, що свідчить про подолання більшістю здобувачів пасивності у професійно-особистісному зростанні. У КГ спостерігаються лише незначні коливання: високий рівень виявлено у 23,53% студентів, середній рівень – у 52,94%, а низький рівень зменшився на 3,6% і становить 23,53%.

Відповідно до завдань дослідження важливими складниками технологій проблемно зорієнтованого навчання було залучення студентів до проблематизації, аналізу проблемних ситуацій та кейсів різного типу, виконання індивідуальних та групових проєктів. Зазначені форми роботи передбачали рефлексивну діяльність у межах конкретного завдання. Так, воркшоп «Ринок праці очима першокурсників: де я зараз і де хочу бути» включає «Етап 4. Рефлексія та формування візії (20 хв). Презентація результатів груп. Формулювання індивідуальних цілей на поточний семестр. Кожен студент записує для себе одну конкретну дію, яку він зробить уже завтра для розвитку», хакатон «Зворотнє проєктування» – «Етап 4. Рефлексія та захист результатів . Кожна команда презентує результати свого дослідження. Фокус презентації. Не просто показати, що «зараз все працює», а пояснити свій методологічний шлях: як саме вони здогадалися про структуру мережі; які хибні гіпотези були відкинута; чому нове архітектурне рішення є оптимальним з точки зору ресурсів та безпеки», Е-портфоліо – детальний рефлексивний критерій оцінювання. На нашу думку, саме тому результати підсумкової діагностики за рефлексивним критерієм засвідчили виразну позитивну тенденцію щодо зростання рівня готовності майбутніх фахівців до неперервної самоосвіти та професійного вдосконалення.

Загальна динаміка показників за рефлексивним критерієм у студентів ЕГ виглядає таким чином: збільшення кількості здобувачів, що виявили високий рівень – на 19.1%, середній – на 2,13%, зменшення на низькому рівні – на 21,28%. Щодо студентів КГ, то виявлена інша тенденція: приріст кількості студентів, що показали високий рівень – на 5,88%, тоді як за середнім рівнем відбулось зменшення на 1,96%, відповідно зменшення за низьким рівнем становить 3,92%. Зіставлення показників свідчить про суттєву динаміку в студентів ЕГ, що зумовлено їх активним залученням до самоаналізу та взаємоаналізу результатів різних форм проєктних завдань, вирішення кейсів, впровадженню практик код-рев'ю, рефлексивних сесій, ведення е-портфоліо.

У межах завдань контрольного етапу дослідно-експериментальної роботи використовувалися такі методи та методики для дослідження рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії за комунікативним критерієм, як тест «Потреба в спілкуванні» (Ю. Орлов) (Додаток Б.12), методика «Комунікативні й організаторські здібності» (В. Синявський, В. Федоришин) (Додаток Б.13), методика діагностики функціонально-рольових позицій у команді (Р. Белбін) (Додаток Б.11), кейс-тести для визначення умінь та навичок комунікації у фаховому та непрофесійному середовищах (Додаток Б.14), спостереження за неформальною комунікацією студентів під час проведення тренінгів, воркшопів та міні-хакатонів.

Самооцінка здобувачами комунікативних навичок в професійному середовищі (Додаток А, запитання 16, 17) засвідчила суттєве зростання показників в ЕГ. Частка студентів із високим рівнем (здатних точно оперувати поняттями та вільно вести фаховий діалог) в ЕГ зросла на 21,28% і сягнула 38,30% (у КГ – лише 19,61%). При цьому кількість студентів із низьким рівнем у ЕГ скоротилася з 34,04% до 8,51%, тоді як у КГ цей показник залишився високим – 31,37%. Зіставлення результати самооцінки студентів щодо їхньої здатності презентувати та публічно захищати розроблені інженерні рішення на контрольному етапі також виявило перевагу студентів ЕГ. Частка студентів ЕГ, які демонструють високий рівень (упевнено захищають проєкт із використанням технічних аргументів, графіків та результатів симуляцій), зросла на 21,90% і досягла 38,30% (проти 19,61% у КГ). Водночас частка студентів, які намагалися уникати публічних виступів через труднощі у структуруванні викладу (низький рівень), в ЕГ скоротилася до 8,51%, тоді як у КГ вона залишається значною – 31,37%.

За результатами діагностики на контрольному етапі за методикою «Потреба в спілкуванні» низький рівень виявлено у 10,64% студентів ЕГ та 31,37% КГ, середній – у 53,2% ЕГ і 49,02% КГ, а високий – у 36,17% ЕГ та 19,6% КГ; за методикою «Комунікативні та організаторські здібності»

низький рівень виявлено у 2,13% студентів ЕГ та 19,61% КГ, середній – у 55,32% ЕГ і 52,94 КГ, а високий досяг 42,55% в ЕГ та 27,45% у КГ.

Аналіз результатів опитування за методикою Р. Белбіна, спрямованої на визначення функціонально-рольових позицій у команді, на контрольному етапі засвідчив позитивну динаміку у студентів експериментальної групи: кількість осіб із високим рівнем зросла до 36,17% (ЕГ) проти 17,65% у КГ, середня група зросла до 51,06% (ЕГ) порівняно з 47,06% у КГ, тоді як частка студентів із низьким рівнем суттєво скоротилася до 12,77% в ЕГ проти 35,29% у контрольній групі. Відзначимо також як позитивний момент структурну оптимізацію рольового профілю учасників командної взаємодії. Замість стихійного переважання комунікативно-соціальних ролей («Генератор ідей / Нетворкер» та «Організатор команди / Помічник»), які раніше гальмували виконання технічних завдань, домінуючий вектор зміщується у бік високого рівня прояву ролей «Мислителя», «Оцінювача ідей», «Спеціаліста» та «Фіналізатора». Це забезпечує баланс між генерацією ідей, їх аналізом, технічним втіленням та якісним завершенням проєктів.

Важливим елементом діагностики слугували кейс-тести для визначення умінь та навичок комунікації у фаховому та непрофесійному середовищах. З урахуванням того, що в процесі навчання у ЗВО студенти набули певного досвіду роботи у таких середовищах, на контрольному етапі дослідження було запропоновано більш складний комплекс кейс-тестів (Додаток Б.14). Наведемо два приклади.

«Кейс-тест. Користувач корпоративної програми (бухгалтер) пише у техпідтримку: «Ваша програма видає помилку "HTTP 500 Internal Server Error" при вивантаженні звітів, ви все зламали, робота стоїть!». Як комп'ютерний інженер техпідтримки, ви маєте відповісти користувачу?». Більшість студентів обрали цілком адекватну відповідь взаємодії в непрофесійному середовищі: «Вітаємо! Нам прикро, що ви зіткнулися з цією проблемою. Наші інженери вже виправляють технічний збій у модулі звітів. Орієнтовний час відновлення – 20 хвилин. Ми повідомимо вас одразу, як усе

запрацює». Доцільність саме такої позиції пов'язана із тим, що користувач, який не має спеціальної технічної освіти, перебуває у стані стресу через зупинку роботи й тому йому потрібна емпатія, чіткий статус простими словами та розуміння термінів вирішення, а не внутрішній технічний жаргон.

«Кейс-тест. Ви виступаєте в ролі системного архітектора. Вам потрібно передати команді розробників вимоги до інтеграції нового периферійного модуля в існуючу комп'ютерну систему». Студенти, які вже мають досвід взаємодії в професійному середовищі, цілком зрозуміло обрали відповідь «Створити детальне технічне завдання (ТЗ) у таск-трекері (наприклад, Jira) з додаванням схем алгоритмів, опису API-ендпоінтів, форматів JSON-запитів та чітких критеріїв приймання (Acceptance Criteria)», оскільки для розробників критично важливо мати формалізовані, однозначні та зафіксовані в таск-трекері технічні параметри, а не усні інструкції чи розмиті описи.

Результати виконання студентами кейс-тестів щодо комунікації на контрольному етапі засвідчили позитивну динаміку в ЕГ: частка студентів із високим рівнем комунікативних умінь та навичок зросла з 10,64% до 34,04% (у КГ – з 11,76% лише до 13,73%), середня група досягла 44,68% в ЕГ проти 45,1% у КГ, тоді як показники низького рівня в ЕГ скоротилися майже удвічі – до 21,28% (проти 39,22% у КГ).

У цілому за результатами аналізу у межах комунікативного критерію зроблено висновки про зростання кількості студентів ЕГ, які показали високий рівень. Натомість в студентів КГ динаміка набагато повільніша. Означена динаміка в ЕГ зумовлена активним включенням студентів у різні форми комунікативної практики під час міні-хакатонів, моделювання ситуацій спілкування, що демонстрували особливості вербальної та письмової комунікації із різними категоріями професіоналів та непрофесіоналів. Крім того, свідомий аналіз умінь та навичок командної роботи, варіантів моделей функційно-рольових позицій у діяльності команд сприяв підвищенню інтересу до можливостей самопрезентації у груповій діяльності, набуттю досвіду комунікативної взаємодії у різних спільнотах.

За результатами дослідно-експериментальної роботи було проведено узагальнення результатів дослідження рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованого навчання. Засвідчена суттєва динаміка у студентів ЕГ. Частка студентів із високим рівнем збільшилась на 19,15% та, відповідно, виявлено, що на 23,4% зменшилась частка студентів із низьким рівнем. У КГ динаміка несуттєва: низький рівень зменшилися на 3,92%, високий збільшився на 3,92% (Рис. 2.4., 2.5, Таблиця 2.5).

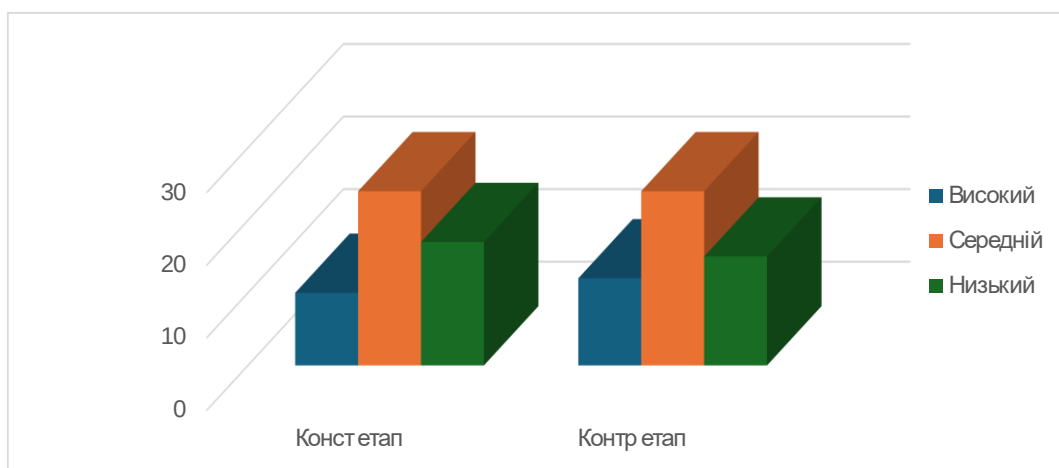


Рис. 2.4. Динаміка сформованості методологічної компетентності студентів контрольної групи

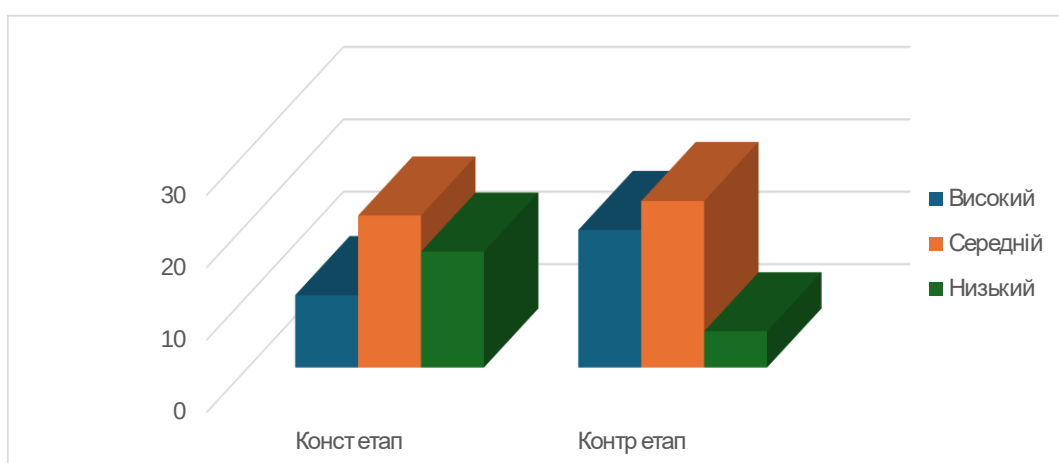


Рис. 2.5. Динаміка сформованості методологічної компетентності студентів експериментальної групи

Динаміка рівнів сформованості методологічної компетентності в студентів ЕГ і КГ за результатами дослідження (осіб)

Рівні	Експериментальна група		Контрольна група	
	Констатувальний етап	Контрольний етап	Констатувальний етап	Контрольний етап
Високий	10	19	10	12
Середній	21	23	24	24
Низький	16	5	17	15

В рамках нашого дослідження будемо використовувати критерій однорідності χ^2 (Пірсона) – це непараметричний статистичний метод, який використовується для перевірки того, чи існує статистично значуща відмінність між розподілами частот у двох незалежних вибірках. Важливо підкреслити, що критерій можливо використовувати тому що усі очікувані частоти більше або дорівнюють 5 (мінімальна очікувана частота становить 9,59 на обох етапах експерименту).

Першочерговим завданням математико-статистичного аналізу в межах експерименту є верифікація стартового стану досліджуваних сукупностей. Перевірка спрямована на підтвердження нульової гіпотези про відсутність значущих відмінностей між контрольною та експериментальною групами, що доводить їхню однорідність за рівнем сформованості методологічної компетентності до моменту впровадження розробленої методики навчання.

Нульова гіпотеза (H_0): Розподіл частот (рівнів компетентності) в експериментальній та контрольній групах не має статистично значущих відмінностей (групи однорідні).

Альтернативна гіпотеза (H_1): Розподіл частот у групах статистично відрізняється.

Визначаємо кількість рівнів свободи $df = (R - 1) \cdot (C - 1) = 2$ Де R – кількість рядків (рівнів = 3), C – кількість стовпців (груп = 2: ЕГ, КГ).

Обираємо рівень значущості $\alpha = 0.05$

Для кількості рівнів свободи 2 та рівня значущості 0.05 знаходимо $\chi^2_{\text{крит}} = 5.99$

Якщо $\chi^2_{\text{емп}} < \chi^2_{\text{крит}}$ – приймається H_0 (відмінностей немає).

Якщо $\chi^2_{\text{емп}} \geq \chi^2_{\text{крит}}$ – приймається H_1 (відмінності статистично значущі).

Класична формула має вигляд

$$\chi^2_{\text{емп}} = \sum \frac{(O_i - E_i)^2}{E_i}$$

Де O_i (Observed) – фактичні частоти;

E_i (Expected) – очікувані частоти.

Формула для обрахунку для випадку коли у нас є дві незалежні вибірки обсягом N_1, N_2 , та f_{1i}, f_{2i} є частотою здобувачів освіти, які отримали відповідний рівень у перший (експериментальній) та у другий (контрольний) набуває вигляду.

$$\chi^2_{\text{емп}} = \frac{1}{N_1 N_2} \cdot \sum_{i=1}^k \frac{(f_{2i} \cdot N_1 - f_{1i} \cdot N_2)^2}{f_{1i} + f_{2i}}$$

Складаємо аналітичну таблицю для констатувального етапу (Таблиця 2.6).

Таблиця 2.6.

Аналітична таблиця для обрахунку $\chi^2_{\text{емп}}$ для констатувального етапу

рівні	f_{1i}	f_{2i}	$(f_{2i} \cdot N_1 - f_{1i} \cdot N_2)^2$	$f_{1i} + f_{2i}$	$\frac{(f_{2i} \cdot N_1 - f_{1i} \cdot N_2)^2}{f_{1i} + f_{2i}}$
Високий	10	10	1600	20	80,0
Середній	21	24	3249	45	72,2
Низький	16	17	289	33	8,8
					161

Отже

$$\chi^2_{\text{емп}} = 0.067$$

Отримуємо $\chi^2_{\text{емп}} < \chi^2_{\text{крит}}$ – приймається H_0 (відмінностей немає), тобто до початку впровадження технологій проблемно зорієнтованого навчання експериментальна та контрольна групи статистично не відрізнялися (були однорідними за рівнем сформованості методологічної компетентності).

Складаємо аналітичну таблицю для контрольного етапу (Таблиця 2.7).

Таблиця 2.7.

Аналітична таблиця для обрахунку $\chi^2_{\text{емп}}$ для контрольного етапу

рівні	f_{1i}	f_{2i}	$(f_{2i} \cdot N_1 - f_{1i} \cdot N_2)^2$	$f_{1i} + f_{2i}$	$\frac{(f_{2i} \cdot N_1 - f_{1i} \cdot N_2)^2}{f_{1i} + f_{2i}}$
Високий	19	12	164025	31	5291,1
Середній	23	24	2025	47	43,1
Низький	5	15	202500	20	10125,0
					15459,2

Отже

$$\chi^2_{\text{емп}} = 6.45$$

$\chi^2_{\text{емп}} \geq \chi^2_{\text{крит}}$ – приймається H_1 (відмінності статистично значущі), після експерименту розподіл у групах став статистично різним, що доводить що групи відрізняються за рівнем сформованості методологічної компетентності.

За результатами проведених розрахунків приймається альтернативна гіпотеза, згідно з якою рівні сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії ЕГ та КГ між собою суттєво відрізняються, отже різниця в структурі статистично значуща.

Узагальнений аналіз отриманих результатів дозволяє зробити висновок, що сформульована на початку дослідження гіпотеза – процес формування методологічної компетентності майбутніх фахівців з

комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання буде більш ефективним, якщо розробити та упровадити відповідні педагогічні умови.

Отримані результати свідчать про статистично значущі кількісні та якісні зміни, що відбулися в структурних компонентах методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

ВИСНОВКИ ДО РОЗДІЛУ 2

Підсумки проведеного дослідно-експериментального дослідження свідчать про ефективність процесу формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

З урахуванням теоретико-методичних засад дослідження, результатів констатувального етапу експерименту визначено та обґрунтовано педагогічні умови, що забезпечують ефективність формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання: актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності

Теоретико-методичне обґрунтування педагогічних умов здійснено з урахуванням наукових положень психології мотивації особистості, мотивації пізнавальної діяльності студентів технічних спеціальностей, психологічних

особливостей мотивації IT-спеціаліста; теоретико-методичних засад професійної підготовки майбутніх фахівців з комп'ютерної інженерії; сутнісних ознак методологічного знання, методологічної підготовки здобувачів вищої освіти в ЗВО; методології комп'ютерної інженерії; теорії та практики розробки та використання проблемних завдань в освітньому процесі вищої школи; діяльнісного, праксеологічного та компетентнісного підходів у вищій освіті; моделей та технологій проблемно зорієнтованого навчання; технологій проблематизації в освіті; кейс-технології як складника проблемно зорієнтованого навчання.

З метою формування мотивації фахівців з комп'ютерної інженерії до усвідомлення цінності методологічного знання у професійній діяльності фахівців IT-сфери розроблено та впроваджено мотиваційний тренінг, симуляційний ІІІ-практикум, інтерактивний воркшоп, використано відео-кейси як дієвий інструмент проблемно зорієнтованого навчання та ін.

Упроваджено комплекс наскрізних проблемно-зорієнтованих сценаріїв, завдань щодо усвідомленого використання загальнонаукових методів пізнання (аналіз, синтез, класифікація, порівняння, узагальнення та ін.).

З метою формування методологічних умінь та навичок, навичок командної роботи здобувачі вищої освіти залучалися до аналізу кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності (архітектурно-структурні, ресурсно-технологічні, інтеграційної та еволюційної сумісності, інженерної безпеки та стійкості), міні-хакатонів.

Аналіз результатів дослідно-експериментальної роботи підтвердив ефективність розроблених педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованих технологій. Динаміка приросту за всіма рівнями в експериментальній групі порівняно з контрольною суттєво відрізняється, що дозволяє говорити про підтвердження висунутої гіпотези дослідження.

ЗАГАЛЬНІ ВИСНОВКИ

У дисертації наведено теоретичне обґрунтування й вирішення наукової проблеми формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами проблемно зорієнтованого навчання, що полягає в обґрунтуванні та експериментальній перевірці педагогічних умов, які забезпечують ефективність досліджуваного процесу.

1. Аналіз філософської та психолого-педагогічної літератури з проблеми дослідження зумовив розгляд поняття «методологія» як вчення про методи пізнання та перетворення дійсності. Акцентовано увагу на значущості методології практичної діяльності як сфери практичної діяльності, спрямованої на створення технологій трансформації, виробництва нового знання та інноваційних ідей з урахуванням специфіки методологічного мислення.

2. З урахуванням наукових положень щодо сутності поняття «компетентність», європейських та вітчизняних нормативних документів щодо компетентнісного підходу в освіті визначено, що компетентність майбутніх фахівців різного профілю розглядається як динамічна комбінація знань, умінь і навичок, способів мислення, професійних, світоглядних і громадянських якостей, морально-етичних цінностей, які визначають здатність успішно здійснювати професійну та навчальну діяльність і є результатом навчання на певному рівні вищої освіти. Доведено, що попри різноманітність формулювань і специфічних акцентів, методологічна компетентність у сучасному науковому вимірі постає не просто як результат засвоєння теорії, а як цілісна, динамічна здатність фахівця творчо осмислювати дійсність, продукувати нове знання та ефективно проєктувати власну професійну діяльність.

На підставі аналізу особливостей умов професійної діяльності фахівців з комп'ютерної інженерії (надшвидка зміна технологічних інновацій, інтеграція штучного інтелекту, розвиток кіберфізичних систем та хмарних обчислень, перехід від пріоритету обчислювальної потужності до парадигми енергоефективності та обмеженості ресурсів та ін.), функцій методологічного складника такої діяльності (когнітивна, нормативно-регулятивна, проєктно-

конструктивна, рефлексивна, прогностична, преадаптивна та комунікативна) методологічна компетентність майбутнього фахівця з комп'ютерної інженерії розглядається як компонент професійної компетентності, інтегративне утворення особистості, що визначає здатність інженера свідомо обирати, комбінувати та ефективно застосовувати фундаментальні методи наукового пізнання, математичного моделювання та системного проєктування для розв'язання складних технічних задач.

Структура досліджуваного виду компетентності представлена у взаємозв'язку та взаємодоповненні таких компонентів, як *мотиваційний* (усвідомлення значущості фундаментальних методологічних знань та умінь для кар'єрного успіху, прагнення до інтелектуальної самореалізації в умовах стрімкого розвитку ІТ-сфери, невизначеності та нестабільності ринку праці), *когнітивний* (система фундаментальних знань про методологію наукового пізнання, теорію систем, методологію комп'ютерно-інженерної діяльності, принципи та логіку побудови обчислювальних процесів), *праксеологічно-технологічний* (володіння загальнонауковими методами пізнання, проблематизації комп'ютерно-інженерної дійсності, методами моделювання, мовами програмування, засобами автоматизації проєктування та діагностики, сформованість навичок системного мислення, толерантність до невизначеності), *рефлексивний* (здатність до самоаналізу, критичного переосмислення методів роботи та готовність до швидкої трансформації професійного досвіду в умовах невизначеності), *комунікативний* (здатність працювати в команді, здатність обирати адекватні методи, стилі та канали вербальної і письмової комунікації залежно від рівня технічної підготовки аудиторії).

Окреслено методологічні підходи, на яких ґрунтується процес формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії – системний, діяльнісний, особистіно зорієнтований, компетентнісний та праксеологічний.

3. Розкрито концептуальні засади проблемно зорієнтованого навчання як актуального напрямку вдосконалення освітнього процесу у вищій школі в цілому

та професійної підготовки майбутніх фахівців ІТ-сфери зокрема. Результати аналізу наукових підходів зарубіжних та вітчизняних авторів щодо сутності, моделей, технологій проблемно зорієнтованого навчання зумовили його розгляд як: навчального підходу, який допомагає студентам розвивати (удосконалювати) гнучкі знання, навички ефективного розв'язання проблем, навички самонавчання і навчання впродовж життя, навички ефективної співпраці, а також внутрішню мотивацію; навчального методу, спрямованого на розвиток у студентів здатності самостійно вирішувати проблеми, самостійно навчатися протягом усього життя та працювати в команді. Доведено, що технології проблемно зорієнтованого навчання передбачають цілеспрямоване використання проблемних завдань різного типу, кейс-методу, проєктного навчання, симуляцій, аналізу відеоресурсів та ін.

4. У дисертаційній роботі розроблено критерії та показники сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії (мотиваційний, когнітивний, діяльнісний, рефлексивний та комунікативний) та відповідні рівні (низький, середній, високий).

Констатувальний етап експериментальної роботи засвідчив переважно середній та низький рівні сформованості методологічної компетентності фахівців означеного профілю.

5. Доведено, що ефективність формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії зумовлена теоретичним обґрунтуванням та впровадженням педагогічних умов, які визначалися на підставі аналізу теоретичних засад дослідження, результатів констатувального етапу експерименту.

Розроблено й упроваджено педагогічні умови формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії: актуалізація мотивації та усвідомлення цінності методологічних знань, умінь та навичок в контексті професійного становлення та кар'єрного зростання майбутніх фахівців; інтеграція фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних

проблемних завдань; включення майбутніх фахівців з комп'ютерної інженерії у квазіпрофесійну діяльність на основі кейсів різного типу, завдань щодо проблематизації комп'ютерної інженерної дійсності, міні-хакатонів з метою формування методологічних умінь та навичок, навичок командної роботи; розвиток рефлексивної позиції майбутніх фахівців на всіх етапах формування методологічної компетентності.

6. Реалізація визначених педагогічних умов здійснювалася в процесі організації аудиторної та позааудиторної роботи з блоку фахових дисциплін відповідно до навчального плану підготовки здобувачів вищої першого (бакалаврського) рівня вищої освіти спеціальності 123 «Комп'ютерна інженерія».

З метою формування мотивації фахівців з комп'ютерної інженерії до усвідомлення цінності методологічного знання у професійній діяльності фахівців ІТ-сфери розроблено та впроваджено мотиваційний тренінг «Керуй алгоритмом: мистецтво методологічного мислення в ІТ», симуляційний ІІІ-практикум «Інженерне мислення: точка зламу», інтерактивний воркшоп «Ринок праці очима першокурсника: де я зараз і де хочу бути», використано відеокейси як дієвий інструмент проблемно зорієнтованого навчання (зокрема виступи провідних експертів галузі на TED-конференціях), симуляційний ІІІ-практикум «Інженерне мислення: точка зламу».

Для інтеграції фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії на основі міждисциплінарних проблемних завдань розроблено та впроваджено комплекс наскрізних проблемно зорієнтованих сценаріїв, завдання на основі загальнонаукових методів пізнання (аналіз, синтез, класифікація, порівняння, узагальнення та ін.), алгоритмічні приписи (правила) використання таких методів.

З метою формування методологічних умінь та навичок, навичок командної роботи здобувачі вищої освіти включалися у квазіпрофесійну діяльність на основі кейсів різного типу (аналітичні, техніко-діагностичні, проєктно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), завдань щодо проблематизації комп'ютерної інженерної дійсності (архітектурно-

структурні, ресурсно-технологічні, інтеграційної та еволюційної сумісності, інженерної безпеки та стійкості), мініхакатонів, зокрема на основі віртуалізованого середовища в симуляторах мережевої інфраструктури (Cisco Packet Tracer / GNS3). З метою формування рефлексивної позиції використано метод портфоліо (накопичення та публічний аналіз власних програмних продуктів, схемотехнічних рішень та архітектурних проєктів тощо); взаєморецензування (в ІТ-сфері – *Code Review*) та ін.

Аналіз результатів формувального етапу експерименту засвідчив позитивні зміни в динаміці рівнів сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії в експериментальних групах. Так, кількість студентів із низьким рівнем сформованості досліджуваного феномену в ЕГ виявилася на 18,77% меншою від КГ, а з високим – на 16,9% більша від КГ.

Підтверджені за допомогою критерію Пірсона позитивні якісні й кількісні зміни є підґрунтям для загального висновку про ефективність запропонованих педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання.

Проведене дослідження не вичерпує всіх аспектів проблеми. Перспективу подальшої роботи вбачаємо в дослідженні питань формування системного мислення фахівців ІТ-сфери в умовах тотального використання штучного інтелекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Баранюк О. Ф. Проблемно-орієнтоване навчання у програмній інженерії. *Наукові записки Кіровоградського державного педагогічного університету імені Володимира Винниченка*. Серія : Проблеми методики фізико-математичної і технологічної освіти. 2016. Вип. 9 (1). С. 3–10.
2. Бех І. Діяльнісний і компетентнісний підходи: сутність та сфери застосування. *Директор школи, ліцею, гімназії*. 2014. № 1-4. С. 28–34.
3. Биков В., Спірін О., Пінчук О. Сучасні завдання цифрової трансформації освіти. *Вісник Кафедри ЮНЕСКО «Неперервна професійна освіта XXI століття»*. 2020. № 1. С. 27–36.
4. Бирка М. Напрями вдосконалення освітнього процесу майбутніх фахівців з комп'ютерної інженерії в закладах вищої освіти. *Українська професійна освіта=Ukrainian Professional Education*. 2019. № (5). С. 97–101.
5. Біда О. А. Сутність та структура дослідницької компетентності майбутніх фахівців. *Наукові записки Центральноукраїнського державного педагогічного університету імені Володимира Винниченка*. Серія : Педагогічні науки. 2017. Вип. 159. С. 10–13.
6. Біліченко О., Семенов О. Формування методологічної культури магістрантів та аспірантів на засадах академічної співпраці: міжрегіональний проєкт. *Професіоналізм педагога: теоретичні й методичні аспекти*. 2025. Вип. 24. С. 5–21.
7. Боднар С. В., Глушко Т. В. Особливості проблемно-орієнтованого навчання учителів іноземних мов. *Академічні студії*. Серія : Педагогіка. 2022. Вип. 4. С. 100–107.
8. Бондаренко Л. І. Формування дослідницької компетентності майбутніх викладачів вищих навчальних закладів в умовах магістратури : дис. ... канд. пед. наук : 13.00.04 / ДЗ «Луганський національний університет імені Тараса Шевченка». Старобільськ, 2015. 247 с.

9. Бондаренко Н.В., Косянчук С.В. Проблематизація як атрактор удосконалення мовної освіти. *Digitalization and information society. Selected issues*. Poland, Katowice, 2022. С. 161–184.
10. Бутенко Л. Л. Технології проблематизації в навчальній та науково-дослідницькій діяльності студентів університету. *Вісн. Луган. нац. ун-ту імені Тараса Шевченка* : Педагогічні науки. 2016. № 1 (298). Ч. 2. С. 20–29.
11. Бутенко Л. Л. Методологізація загальнопедагогічної підготовки майбутніх учителів та науково-педагогічних працівників в умовах трирівневої системи вищої освіти: дис. д-ра пед. наук : 13.00.04 / ДЗ «Луганський національний університет імені Тараса Шевченка». Старобільськ, 2021. 590 с.
12. Бутенко Л. Л., Шехавцова С. О. Взаємозв'язок методологічної компетентності та неявного знання у професійній діяльності викладача вищої школи. *Вісник науки та освіти*. 2026. № 5 (47). С. 2716–2731.
13. Ваховський Л. Ц. Постнекласична наука й перспективи розвитку університетської освіти. *Вісн. Луган. нац. ун-ту імені Тараса Шевченка*. Педагогічні науки. 2019. № 6 (329), Ч. II. С. С. 158–165.
14. Великий тлумачний словник сучасної української мови; уклад. і голов. ред. В. Т. Бусел. Київ ; Ірпінь : ВТФ «Перун», 2009. 1736 с
15. Вихрущ В. О., Козловський Ю. М. Методологія та методика наукового дослідження. Львів : Видавництво Львівської політехніки. 2020. 336 с.
16. Вихрущ А. В. Методологічна культура здобувачів у закладах вищої освіти. *Педагогічна наука і освіта XXI століття*. 2025. № 4. С. 17–30.
17. Вінник М. О. Формування науково-дослідницької компетентності майбутніх інженерів-програмістів в умовах освітнього середовища вищого навчального закладу: дис. ... канд. пед. наук : 13.00.01 / Херсонський державний університет. Харків, 2016. 247 с.
18. Волкова Н. П. Інтерактивні технології навчання у вищій школі: навч.-метод. посіб. Дніпро : Університет імені Альфреда Нобеля, 2018. 360 с.

19. Гаврілова Л. Г., Топольник Я. В. Цифрова культура, цифрова грамотність, цифрова компетентність як сучасні освітні феномени. *Педагогічні науки: Інформаційні технології і засоби навчання*. 2023. № 8-9. С.132–133.
20. Галан Л. В., Борисевич Є. Г. Тренінг-курс з організації командної роботи: навч. посіб. Одеса: ДУІТЗ; Одеса: ФОП Бондаренко М.О., 2024. 100 с.
21. Гапоненко Л.О. Рефлексія як умова розвитку особистості : монографія. Дніпропетровськ : Пороги, 2009. 175 с.
22. Гарко І. І., Пирог М. В., Міронова В. Л. Застосування AGILE-методології у навчанні алгоритмізації і основ програмування студентів спеціальності комп'ютерні науки. *Інформаційні технології і засоби навчання*, 2021, Том 85, №5. С. 147–162.
23. Гладун А. Я., Пучков О. О., Субач І. Ю., Хала К. О. Англо-український словник термінів з інформаційних технологій та кібербезпеки. Київ : ІСЗІ КПІ ім. Ігоря Сікорського, 2018. 380 с.
24. Голибард Є. Основи технології системного мислення : підручник. Київ : Фенікс, 2017. 320 с.
25. Гончаренко С. Український педагогічний словник. Київ : Либідь, 1997. 376 с.
26. Гончаренко С. У. Фундаменталізація професійної освіти як дидактичний принцип. *Теорія і практика управління соціальними системами: філософія, психологія, педагогіка, соціологія*. 2008. Вип. 2. С. 87–91.
27. Гончаренко С. У. Педагогічні дослідження : методологічні поради молодим науковцям. Київ ; Вінниця ; ТОВ Фірма «Планер», 2010. 308 с.
28. Гончаренко С. У. Формування у дорослих сучасної наукової картини світу: монографія. Київ: ІПОД НАПН України, 2013. 220 с.
29. Гречаний О. Ф., Сабадуха В. О. Філософія здібностей у контексті пріоритету духовного над матеріальним: монографія. Луганськ : Вид-во СЛУ ім. Володимира Даля, 2015. 211 с.
30. Діяльнісні засади підготовки майбутніх компетентних фахівців в умовах сучасних викликів : монографія / Наукова редакція Олександра

Дубасенюк, Зоя Шарлович. Ломжа – Житомир, Вид-во MANS w Łomży ; ЖДУ імені І.Франка, 2024. 366 с.

31. Добронравова І. Практична філософія науки. Суми : Університетська книга, 2017. 352 с
32. Додонов О. Г., Сенченко В. Р., Коваль О. В. Аналітика та знання в комп'ютерних системах: монографія. 2-ге вид. переробл. та допов. Київ : НАН України, Інститут проблем реєстрації інформації НАН України, Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського», 2025. 326 с.
33. Енциклопедія освіти / Акад. пед. наук України ; головний ред. В. Г. Кремень. Київ : Юрінком Інтер, 2008. 1040 с.
34. Єршов М.-О. Тенденції розвитку ІТ-освіти в Незалежній Україні: монографія. Київ: «Видавництво Людмила», 2023. 350 с.
35. Закон України «Про вищу освіту» від 01.07.2014 № 1556-VII. Законодавство України / Верховна Рада України. URL: <https://zakon.rada.gov.ua/laws/show/1556-18>
36. Закон України «Про освіту» від 5 вересня 2017 року № 145-VIII. URL: <http://zakon4.rada.gov.ua>
37. Занюк С. С. Психологія мотивації та емоцій: навч. посіб. Луцьк : Вид-во Волин. держ. ун-ту ім. Лесі Українки, 1997. 180 с.
38. Зливков В. Л., Лукомська С. О. Сучасні тренінгові технології розвитку особистості в освіті. Київ, 2022. 184 с.
39. Зубик Л. В. Формування професійних компетентностей майбутніх бакалаврів з інформаційних технологій у процесі вивчення фахових дисциплін дис. ... канд. пед. наук: 13.00.04 / Національний університет водного господарства та природокористування. Київ, 2016. 358 с.
40. Інноваційні освітні технології : навчально-методичний посібник / упорядник Л. М. Прокопів. Івано-Франківськ, 2020. 172 с.

41. Інформаційні технології у вищій школі : монографія / за заг. ред. Вакалюк Т. А., Литвинової С. Г. Житомир : Вид-во ФОП "О.О.Євенок", 2019. 364 с.
42. Інформаційно-комунікаційні технології в освіті: словник. Київ : ЦП Компринт, 2019. 134 с.
43. Карташова Ж. Ю., Кузів М. В., Задоріна О. М. Ефективність проблемно-орієнтованого навчання у підвищенні активності здобувачів вищої освіти. *Global Innovations and Collaborative Solutions in Contemporary Science* : International scientific conference (December 15, 2023, Poland). 2023. С. 25–28.
44. Кисельова О. І., Коломієць Л. В., Шевцов А. Г. Якість вищої освіти: організація навчання та вимірювання знань: монографія. Одеса : Бондаренко М.О., 2017. 244 с.
45. Кісіль З. Р., Швець Д. В. Мотивація діяльності людини: навчальний посібник у схемах, таблицях, коментарях. Одеса : Видавництво ОДУВС, 2023. 154 с.
46. Ключко В. І., Коломієць А. А. Формування мотивації навчально-пізнавальної діяльності студентів технічних спеціальностей : монографія. Вінниця : ВНТУ, 2012. 188 с.
47. Коваленко І. І., Бідюк П. І., Гожий О. П. Вступ до системного аналізу : навчальний посібник. Миколаїв : МДГУ ім. Петра Могили, 2004. 148 с.
48. Коваленко Н. В. Професійна преадаптація майбутніх учителів. *Вісник післядипломної освіти*. 2009. Вип. 11(2). С. 161–168.
49. Коваленко О. Є., Волошин С. М., Гусев Б. С., Нікітенко Є. В., Матієвський В. В. Комп'ютерна схемотехніка. Частина 2 : навчальний посібник. Київ : Компринт, 2023. 331 с.
50. Ковальчук Л. Формування методологічної культури здобувачів вищої освіти. *Вісник Львів. університету*. Серія педагогічна. 2025. Вип. 42. С. 227–240.
51. Код економіки 2025: як ІТ формує економіку України / DataDriven, Асоціація «ІТ Ukraine». Київ, 2025. 111 с. URL:

<https://itukraine.org.ua/files/the-code-economy-2025.pdf> (дата звернення: 05.05.2026).

52. Козлик І. Проблематизація у площині структури наукового предмета (лекційні матеріали до теми). *Султанівські читання*. 2023. Вип. 12. С. 67–73.
53. Козубцов І. Теорія та практика розвитку методологічної компетентності в аспірантів: аналіз науково-педагогічної літератури. *Збірник наукових праць Уманського державного педагогічного університету імені Павла Тичини*. 2014. Вип. 3. С. 154–161.
54. Колесник Ю. М., Авраменко М. О., Моргунцова С. А., Юрченко І. О. Досвід запровадження моделі проблемно-орієнтованого навчання у Запорізькому державному медичному університеті. *Медична освіта*. 2016. № 2. С. 90–92.
55. Колотило М. Філософія в технічному університеті: обґрунтування, легітимація, апологія. *Філософія освіти*. 2016. № 1 (18). С. 150–162.
56. Компетентнісний підхід у вищій освіті: світовий досвід : монографія. Київ : КНЕУ, 2016. 61 с.
57. Компетентнісний підхід у вищій школі: теорія та практика : монографія / кол. авт. ; за заг. ред. О. А. Жукової, А. І. Комишана. Харків : ХНУ імені В. Н. Каразіна, 2021. 264 с.
58. Компетентнісний підхід у сучасній освіті: світовий досвід та українські перспективи / за ред. О. В. Овчарук. Київ : «К.І.С.», 2004. 111 с.
59. Концедайло В. В. Застосування ігрових симуляторів у формуванні професійних компетентностей майбутніх інженерів-програмістів : дис. ... канд. пед. наук (доктора філософії) : 13.00.10 – інформаційно-комунікаційні технології в освіті (011 – Освітні, педагогічні науки) / Житомирський державний університет імені Івана Франка, Інститут інформаційних технологій та засобів навчання НАПН України. Київ, 2018. 290 с.
60. Котенко Н. О. Розвиток дослідницької компетентності магістрів спеціальності інженерія програмного забезпечення. *Перспективи та інновації науки*. 2025. № 3. С. 605–614.

61. Кравцов В. Методологічна компетентність як показник методологічної культури майбутнього вчителя. *Наукові записки Кіровоградського державного педагогічного університету імені Володимира Винниченка*. Серія : Педагогічні науки. 2010. Вип. 91. С. 111–116.
62. Круглик В.С. Система підготовки майбутніх інженерів-програмістів до професійної діяльності у вищих навчальних закладах : дис. ... д-ра пед. наук : 13.00.04 / Запорізький національний університет. Запоріжжя, 2018. 538 с.
63. Кубаєвський М., Фурман А. Методологування як засіб розуміння смислу. *Психологія і суспільство*. 2010. № 4. С. 47–57.
64. Кугай Н. До питання про методологічну компетентність майбутнього вчителя математики. *Молодь і ринок*. 2014. № 11. С. 165–169.
65. Кугай Н. В. Формування методологічних знань і вмінь майбутніх учителів математики у процесі навчання дисциплін математичного циклу : дис. ... д-ра пед. наук / 13.00.02 / Національний педагогічний університет імені М. П. Драгоманова. Київ, 2019. 595 с.
66. Кудріна Т. С. Психологія мотивації: підручник. Київ : Видавничо-поліграфічний центр «Київський університет», 2008. 239 с.
67. Кузнецов О. А. Методологічні особливості викладання курсу філософії у вищих технічних навчальних закладах України. *Актуальні проблеми філософії та соціології*. 2025. № 52. С. 83–88.
68. Кушнір Л. О., Кібенко Л. М., Горячок І. В. Діяльнісний підхід при організації змішаного навчання здобувачів освіти. *Науковий часопис НПУ імені М. П. Драгоманова*. 2013. Вип. 91. С. 141–144.
69. Лаврентьєва О. О. Розвиток методологічної культури майбутнього вчителя природничих дисциплін у процесі професійної підготовки: теоретико-методичний аспект : монографія. Київ : КНТ, 2014. 456 с.
70. Лаврентьєва О. О. Теоретичні і методичні засади розвитку методологічної культури майбутніх учителів природничих дисциплін у процесі професійної підготовки: дис. ... д-ра пед. наук: 13.00.04 / Інститут педагогічної освіти і освіти дорослих. Київ, 2015. 530 с.

71. Лазаренко Н. І. Методологічна культура викладача вищої школи: дефінітивний і структурний аналіз. *Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців: методологія, теорія, досвід, проблеми*, 2016. Вип. 44. С. 115–118.
72. Леврінц М. І., Варга Н. І., Попик Ю. С. Розвиток дослідницької компетентності майбутніх педагогів у вищій школі: зарубіжний досвід. *Науковий вісник Ужгородського університету. Серія: «Педагогіка. Соціальна робота»*. 2023. № 2 (53). С. 71–75.
73. Литвин А. В. Методологічні засади поняття «педагогічні умови» : практ. посібник. 2-е вид., доп. і перероб. Львів : ЛДУБЖД, 2018. 88 с.
74. Лі Ліцюань Структура методологічної компетентності майбутнього вчителя музичного мистецтва. *Інноваційна педагогіка*. 2019. Вип. 14(2). С. 87–91.
75. Лі Ліцюань. Формування методологічної компетентності майбутніх учителів музичного мистецтва в процесі вокальної підготовки : дис. ... канд. пед. наук : 13.00.02 / Державний заклад «Південноукраїнський національний педагогічний університет імені К. Д. Ушинського; Сумський державний педагогічний університет імені А. С. Макаренка. Суми, 2020. 260 с.
76. Лопатюк О. Феномен «професійна рефлексія» як предмет міждисциплінарного дослідження. *Педагогічні науки*. 2017. Вип. 69. С. 79–84.
77. Луговий В. І. Європейська концепція компетентнісного підходу у вищій школі та проблеми її реалізації в Україні. *Педагогіка і психологія*. 2009. № 2. С. 13–26.
78. Лузан П., Тітова О. Розвиток дослідницької компетентності майбутніх докторів філософії: технологічний підхід. *Педагогічні науки: теорія, історія, інноваційні технології*. 2025. № 1 (141). С. 121–141.
79. Луценко Г. В., Козуля Л. В. Аналіз особливостей впровадження проблемно-орієнтованого навчання у системі вищої освіти. *Вісник Чернігівського національного педагогічного університету. Серія : Педагогічні науки*. 2016. Вип. 138. С. 91–95.

80. Макаренко Л. Л., Шпильовий Ю. В. Педагогічні умови формування професійної компетентності фахівців з комп'ютерної інженерії в умовах цифровізації освіти. *Науковий часопис Українського державного університету імені Михайла Драгоманова*. Серія 5. Педагогічні науки: реалії та перспективи. 2025. Вип. 103. С. 80–83.
81. Максименко С. Д., Меєрович М. Й., Шрагіна Л. І. Системне мислення: формування і розвиток : навч. посіб. Київ : Києво-Могилянська академія, 2020. 248 с.
82. Малихін О. В., Рубінська Б. І. Про потенціал формування методологічної компетентності в галузі підготовки кваліфікованих спеціалістів гуманітарних спеціальностей. *Збірник наукових праць Херсонського державного університету. Педагогічні науки*. 2018. Вип. 82(2). С. 176–181.
83. Матієвський В. В. Проблемно зорієнтоване навчання у вищій школі: сучасні підходи, моделі та методи. *Освіта та педагогічна наука*. 2025. № 2(189). С. 68–78.
84. Матієвський В. В. Методологічна компетентність майбутніх фахівців з комп'ютерної інженерії: сутність, функції, структура. *Вісник науки та освіти*. 2026. № 4 (46). С. 4026–4040.
85. Матієвський В. В. Мотиваційний компонент методологічної компетентності майбутніх фахівців з комп'ютерної інженерії: сутність та особливості формування у процесі професійної підготовки. *Педагогічна академія: наукові записки*. 2026. № 29. DOI: <https://doi.org/10.5281/zenodo.20577833>
86. Матієвський В. В. Аналітичні уміння та навички як складник методологічної компетентності майбутніх фахівців з комп'ютерної інженерії. *Інноваційні тенденції сьогодення в сфері природничих, гуманітарних та точних наук: збірник наук. праць з матеріалами VIII Міжнар. наук. конф., м. Тернопіль, 20 лютого, 2026 р. / Міжнар. центр наук. досліджень*. Вінниця : ТОВ «УКРЛОГОС Груп», 2026. С. 247–251.

87. Матієвський В. В. Використання кейс-методу як інструменту проблемно зорієнтованого навчання у підготовці майбутніх фахівців з комп'ютерної інженерії. *Теорія і практика сучасної науки та освіти: матеріали XVIII Міжнар. наук.-практ. конф. м. Львів, 9-10 березня 2026 року. Львів : Львівський науковий форум, 2026. С. 13–16.*
88. Матієвський В. В. Методологічна компетентність майбутніх фахівців з комп'ютерної інженерії в умовах цифровізації освіти. *Плекаємо особистість: наук.-метод. альманах. Вип. 11. Лубни: Держ. закл. «Луган. нац. ун-т імені Тараса Шевченка», 2026. С.110–114.*
89. Матієвський В. В. Форми та методи проблемно зорієнтованого навчання у професійній підготовці майбутніх фахівців з комп'ютерної інженерії. *Recent Trends in Science: Proceedings of the 5th International Scientific and Practical Internet Conference, May 14-15. Dnipro : FOP Marenichenko V.V., 2026. Р. 171–173.*
90. Матієвський В. В. Преадаптаційна функція методологічної підготовки майбутніх фахівців з комп'ютерної інженерії. *Наука і освіта в глобальному та національному вимірах: виклики, загрози, перспективи розвитку : зб. наук. праць за матер. V міжнародної наук.-практ. конф. (Лубни, 28-29 травня 2026 р.) / за ред. Я. І. Юрків, Н. О. Островської. Лубни : ДЗ «Луганський національний університет імені Тараса Шевченка», 2026. С. 111–114.*
91. Меєрович М. Й., Шрагіна Л. І. Технологія формування системного мислення: практичний посібник. Одеса : Бондаренко М. О., 2024. 320 с
92. Мельничук І.М. Праксеологізація як ключовий механізм модернізації вищої освіти в сучасних умовах. *Науковий часопис Національного педагогічного університету імені М. П.Драгоманова. Серія 5. Педагогічні науки: реалії та перспективи. 2021. Вип. 80. Том 2. С. 46–50.*
93. Моделювання компетентнісної професійної освіти в контексті євроінтеграції: монографія / кол. авт; за заг. ред. проф. Н. П. Волкової. Дніпро : Університет імені Альфреда Нобеля, 2021. 356 с.

94. Модернізація освіти в цифровому вимірі: монографія / за наук. ред. Н. Морзе, О. Буйницької. Київ : Київ. ун-т ім. Б. Грінченка, 2021. 300 с.
95. Молодь на ринку праці: навички ХХІ століття та побудова кар'єри. Інформаційно-аналітичні матеріали. Київ : Мін-во молоді та спорту України, Держ. ін-т сімейної та молодіжної політики, 2019. 42 с.
96. Морзе Н., Буйницька О. Формування компетентностей майбутніх фахівців з програмної інженерії. *Інформатика та інформаційні технології в навчальних закладах*, 2015. № 2-3 (56). С. 15–20.
97. Національний освітній глосарій: вища освіта (4-е вид., перероб. і доп.) / Авт.-уклад.: В.Є. Бахрушин, М.І. Винницький, В.М. Захарченко та ін. / За ред. В.Г. Кременя, В.Є. Бахрушина, Ю.М. Рашкевича. Київ, 2024. 114 с.
98. Національний освітньо-науковий глосарій. Київ : ТоВ «КОНВі ПРІНТ», 2018. 524 с.
99. Нікітіна І. В. Розвиток мотивації студентів. Практикум : навч. посіб. Київ : НУХТ, 2017. 326 с.
100. Нічуговська Л. І. Розвиток методологічної компетентності аспірантів в умовах університетського середовища. *Вісник Черкас. нац. університету імені Богдана Хмельницького*. Серія Педагогічні науки. 2024. № 2. С. 38–43.
101. Новий тлумачний словник української мови : у 3 т. Т. 3 / уклад. В. Яременко, О. Сліпущко. Київ : Аконіт, 2005. 863 с.
102. Нужин О. Застосування комплексу інтерактивних групових методів навчання у процесі формування готовності ІТ-фахівців до професійної діяльності. *Освіта. Інноватика. Практика*. 2025. Том 13, № 7. С. 90–96.
103. Ожиндович Л. М. Застосування технологій проблемно/проектно орієнтованого навчання на прикладі курсу "Системи керування вмістом веб-сайтів" для студентів технічних та ІТ-спеціальностей. *Наукові записки Центрально-українського державного педагогічного університету імені Володимира Винниченка*. Серія : Педагогічні науки. 2020. Вип. 185. С. 224–228.

104. Олійник І. В. Основні форми та методи формування у майбутніх докторів філософії дослідницької компетентності в умовах аспірантури. *Перспективи та інновації науки*. Серія «Педагогіка». 2022. № 9 (14). С. 307–322.
105. Опачко М. В. Формування методологічної компетентності майбутнього вчителя фізики у системі професійної підготовки. *Вісник Львівського університету. Серія педагогічна*. Вип. 25. Львів : ЛНУ ім. Івана Франка, 2009. С. 271–278.
106. Основи комп'ютерної інженерії: навчальний посібник [для здобувачів першого (бакалаврський) рівня вищої освіти галузі знань F «Інформаційні технології»] / Уклад.: Л.В. Бубенцова, О.В. Цира. Одеса: ДУІТЗ, 2025. 53 с.
107. Основи наукових досліджень. Конспект лекцій: навч. посіб. для студ. спеціальності 123 «Комп'ютерна інженерія», освітньої програми «Системне програмування та спеціалізовані комп'ютерні системи» / В.П.Тарасенко ; КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, 2022. 55 с.
108. Павленко В. В. Теоретичні основи застосування проблемних ситуацій в освітньому процесі вищої школи. *Актуальні проблеми управління освітою і навчальними закладами* : зб. наук. пр. / за заг. ред. В. Ф. Русакова, І. М. Зарішняк. Вип. 4. Вінниця : ДонНУ імені Василя Стуса, 2021. С. 99–106.
109. Петренко С. Концепція дослідження професійної компетентності ІТ-фахівців. *Педагогічна наука і освіта XXI століття*. 2025. № 4. С. 211–219.
110. Плачинда Т. С. Професійна рефлексія як умова становлення майбутнього фахівця. *Педагогіка вищої та середньої школи*. 2015. № 45. С. 68–72.
111. Поворознюк А. І., Панченко В. І., Філатова Г. Є. Методологія та організація наукових досліджень : навч. посібник ; Нац. техн. ун-т «Харків. політехн. ін-т». Харків : НТУ "ХПІ", 2016. 192 с.
112. Постова С., Опанасюк Н., Опанасюк А. Кейс-метод у цифровому освітньому середовищі: виклики та можливості. *Наукові праці Міжрегіональної академії управління персоналом*. Педагогічні науки. 2025. Вип. 1. С. 18–24.

113. Прилипко В. М., Сохань І. В., Матієвський В. В. Інноваційні методики у вищій освіті як відповідь на виклики війни в Україні. *Академічні візії*. 2024. Вип. 32. DOI: <https://doi.org/10.5281/zenodo.12531404>
114. Проблемно-орієнтоване навчання у викладанні англійської мови як іноземної: теоретичні і практичні питання) : монографія / за заг. ред. Н. Є. Дмитренко. Вінниця : ФОП Т. П. Барановська, 2017. 164 с.
115. Проскура С. Л., Литвинова С. Г. Формування професійної компетентності майбутніх бакалаврів комп'ютерних наук. *Фізико-математична освіта*. 2019. Вип. 2 (20). С. 137–147.
116. Проскура С. Л. Модель формування професійної компетентності майбутніх бакалаврів комп'ютерних наук. *Фізико-математична освіта*. 2019. Випуск 3(21). С. 104–112.
117. Психологічна діагностика мотивації особистості до навчання в умовах інформаційного суспільства : монографія / Н. В. Пророк, Л. О. Кондратенко, Л. М. Манилова та ін. ; за ред. Н. В. Пророк. Київ : Видавничий Дім «Слово», 2020. 131 с.
118. Пустовойт Б. А., Федяй І. О. Кейс-технологія як один із сучасних методів викладання у закладах вищої освіти для формування компетентності майбутніх фахівців. *Наукові записки кафедри педагогіки*. 2018. Вип. 43. С. 401–408.
119. Романовська Л. І. Праксеологічний підхід: теоретико-методологічні аспекти. *Інноваційна педагогіка*. 2020. Вип. 25. Т.2. С. 157–161.
120. Романовський О., Лапузіна О. Проблемно-орієнтоване навчання та його роль у процесі підготовки майбутніх лідерів-професіоналів у технічних університетах України. *Теорія і практика управління соціальними системами*. 2022. № 4. С.3–16.
121. Сабадош Ю. Г. Розвиток самоосвітньої компетентності майбутніх інженерів комп'ютерних систем і автоматики : дис. ... доктора філософії : 015 «Професійна освіта (за спеціалізаціями)» (01 Освіта/Педагогіка) / Вінницький національний технічний університет. Вінниця, 2021. 254 с.

122. Савицька О. В. Соціально-психологічні умови становлення методологічної компетентності студентів. *Nauka i edukacja w warunkach zmian cywilizacyjnych*: Mater. I Międz. Konf. Nauk.-Prakt. / Pod red. M.Andrzejewskiego. Łódź : Nowa nauka, 2019. С. 72–75.
123. Сапожников С. В. Теоретико-методологічні орієнтири формування дослідницької компетентності майбутніх фахівців у системі вищої освіти України. *Педагогіка формування творчої особистості у вищій і загальноосвітній школах*. 2019. № 65 (2). С. 112–118.
124. Сафонова І. О. Методологічна культура педагога-дослідника в аспекті гуманітарності. *Духовність особистості: методологія, теорія і практика*. 2016. № 5 (74). С. 179–192.
125. Селіванов В. Л., Верба О. А., Васильєва М. Д. Методика та оптимізація наукових досліджень : навч. посіб. для студ. освітньої програми «Комп'ютер-ні системи та мережі» за спеціальністю 123 «Комп'ютерна інженерія» / КПІ ім. Ігоря Сікорського. Київ : КПІ ім. Ігоря Сікорського, 2024. 155 с.
126. Семенов М. А. Реалізація P2P моделі цифрового навчання в Lms Moodle. *Відкрите освітнє е-середовище сучасного університету: електронне наукове фахове видання*. 2025. № 19. С. 165–176.
127. Семенов М. А., Донченко В. Ю., Донченко С. М. Застосування гейміфікації та бізнес-симуляцій при стартап-орієнтованому навчанні студентів ІТ-спеціальностей. *Освіта та педагогічна наука*. 2025. № 1 (188). С. 59–71.
128. Семеріков С. О. Фундаменталізація навчання інформатичних дисциплін у вищій школі : монографія ; науковий редактор академік АПН України, д. пед. н., проф. М. І. Жалдак. Кривий Ріг : Мінерал ; Київ : НПУ ім. М. П. Драгоманова, 2009. 340 с.
129. Семигіна Т. В. Застосування проблемно-орієнтованого навчання у прикладних політологічних дисциплінах. *Науковий часопис НПУ імені М. П. Драгоманова*. Серія 22 : Політичні науки та методика викладання соціально-політичних дисциплін. 2014. Вип. 15. С. 184–189.

130. Синельник І. В., Синельник О. В. Інформаційно-комунікаційні технології навчання фізики майбутніх фахівців з комп'ютерної інженерії в умовах віддаленого навчання. *Проблеми інженерно-педагогічної освіти*. 2023. № 81. С. 104–113.
131. Сисоєва С. О., Кристопчук Т. Є. Методологія науково-педагогічних досліджень: підручнику. Рівне : Волинські обереги, 2013. 360 с.
132. Сисоєва С. О., Козак Л. В. Розвиток дослідницької компетентності викладачів вищої школи: навчальний посібник / Київ. ун-т ім. Б. Грінченка. Київ : ТОВ «Видавниче підприємство «ЕДЕЛЬ-ВЕЙС», 2016. 156 с.
133. Системний підхід у вищій школі : навч. посіб. / автори-упоряд. Кочубей Т. Д. , Іващенко К. В. Умань : ПП Жовтий О. О., 2014. 131 с.
134. Ситнік Т. Використання кейс-технології у контексті інноваційного навчання у закладах вищої освіти. *Молодь і ринок*. 2021. № 10. С. 69–73.
135. Ситуаційна методика навчання: теорія і практика / Упор. О. Сидоренко, В. Чуба. Київ : Центр інновацій та розвитку, 2001. 256 с.
136. Сікора Я. Б. Теоретико-методичні засади адаптивної системи професійної підготовки майбутніх фахівців з інформаційних технологій в умовах цифровізації: дис. ... д-ра пед. наук : 13.00.04 / Житомирський державний університет імені Івана Франка. Житомир, 2025. 709 с.
137. Слободянюк Л. В., Чумак М. Є. Структурно-функціональний підхід при формуванні професійної компетентності майбутніх бакалаврів з комп'ютерної інженерії. *Науковий часопис Українського державного університету імені Михайла Драгоманова*. Серія 5. Педагогічні науки: реалії та перспективи. 2025. Вип. 103. С. 102–107.
138. Слободянюк Л. В. Формування професійної компетентності майбутніх бакалаврів з комп'ютерної інженерії у закладах фахової передвищої освіти: дис. ... доктора філософії : 011 Освітні, педагогічні науки / Український державний університет імені Михайла Драгоманова. Київ, 2025. 221 с.
139. Словник термінів ІТ і комп'ютерної інженерії / В. В. Гнатушенко, Г. М. Коротенко, В. І. Олевський та ін.; за ред. В. В. Гнатушенка, Г. М. Коротенка

- Л. І. Цвіркуна. Дніпро : Національний технічний університет «Дніпровська політехніка», 2025. 709 с.
140. Словник-довідник з професійної педагогіки / за ред. А. В. Семенової. Одеса : Пальміра, 2006. 272 с.
 141. Собченко Т., Кін О., Кириленко С. Цифрова компетентність як складова готовності майбутніх фахівців до неперервного професійного розвитку. *Суспільство та національні інтереси*. 2024. №8 (8). DOI: [https://doi.org/10.52058/3041-1572-2024-8\(8\)-350-359](https://doi.org/10.52058/3041-1572-2024-8(8)-350-359)
 142. Сорокопуд М. Дослідницькі компетентності бакалаврів з комп'ютерної інженерії у навчанні фізики. *Наукові записки. Серія: Проблеми методики фізико-математичної освіти*. 2016. Т. 3. № 10. С. 86–90.
 143. Стандарт вищої освіти за спеціальністю 123 «Комп'ютерна інженерія» галузі знань 12 «Інформаційні технології» для першого (бакалаврського) рівня вищої освіти. Київ, 2018. URL: <https://mon.gov.ua/static-objects/mon/sites/1/vishcha-osvita/zatverdzeni%20standarty/123-kompyuterna-inzheneriya.pdf>
 144. Стрюк К. М. Зміст професійної компетентності майбутніх фахівців із комп'ютерної інженерії. *Збірник наукових праць Херсонського державного університету*. Педагогічні науки. 2016. Вип. 73(2). С. 118–122.
 145. Стрюк К. М. Формування професійної компетентності молодших спеціалістів з комп'ютерної інженерії у радіотехнічних коледжах: дис. ... канд. пед. наук (доктора філософії): 13.00.04 / Харк. нац. університет внутр. справ, Українська інженерно-педагогічна академія. Харків, 2020. 334 с.
 146. Сурмін Ю. П. Кейс-метод: становлення та розвиток в Україні. *Вісник нац. академії державного управл. при Президентові України*. 2015. № 2. С. 19–28.
 147. Сучасний психолого-педагогічний словник / авт. кол. за заг. ред. О. І. Шапран. Переяслав-Хмельницький (Київська область) : Домбровська Я.М., 2016. 473 с.
 148. Тверезовська Н., Філіппова Л. Сутність та зміст поняття «педагогічні умови». *Нова пед. думка*. 2009. № 3. С. 90–92.

149. Теоретико-методологічні засади інформатизації освіти та практична реалізація інформаційно-комунікаційних технологій в освітній сфері України : монографія / В. Ю. Биков, О. Ю. Буров, А. М. Гуржій, М. І. Жалдак, М. П. Лещенко, С. Г. Литвинова, В. І. Луговий, В. В. Олійник, О. М. Спірін, М. П. Шишкіна / наук. ред. В. Ю. Биков, С. Г. Литвинова, В. І. Луговий. Київ : Компринт, 2019. 214 с.
150. Топольник Я. Особливості вивчення навчальної дисципліни «Методологія наукових досліджень» в процесі підготовки майбутніх докторів філософії. *Гуманізація навчально-виховного процесу*. 2022. № 1 (101). С. 16–23.
151. Топольник Я. Використання цифрових освітніх середовищ у професійній підготовці майбутніх викладачів закладів вищої освіти. *Гуманізація навчально-виховного процесу*. 2025. № 2 (108). С. 178–186.
152. Топольник Я. В. Епістемологічні засади формування методологічної культури майбутніх викладачів вищої школи. *Перспективи та інновації науки*. 2026. № 6(64). С. 1077–1089.
153. Тулашвілі Ю. Й. Інтеграція фундаментального та праксеологічного підходів як основа формування готовності майбутніх ІТ-фахівців до розв'язання складних професійних завдань. *Сучасні інформаційні технології та інноваційні методики навчання в підготовці фахівців: методологія, теорія, досвід, проблеми*. 2018. Вип. 51. С. 375–379.
154. Хриков Є. М. Педагогічні умови в структурі наукового знання. *Шлях освіти*. 2011. № 2. С. 11–15.
155. Федів В. І., Олар О. І., Бірюкова Т. В. Історичний аспект виникнення і поширення кейс-методу. *Народна освіта*. 2020. Вип. 2. С. 68–72.
156. Філоненко Д. Психологічні особливості мотивації ІТ-спеціаліста. *Збірник наукових праць Національної академії державної прикордонної служби України*. Серія: Психологічні науки. 2018. № 1(9). С. 240–254.
157. Філософський енциклопедичний словник / В. І. Шинкарук (гол. редкол.) та ін. Київ : Абрис, 2002. 742 с.

158. Фролова М. Педагогічні умови формування дослідницьких компетентностей майбутніх спеціалістів з економічної кібернетики. *Молодь і ринок*. 2012. № 9 (92). С. 129–134.
159. Фурман А. В. Теорія навчальних проблемних ситуацій: монографія. Тернопіль: Астон, 2007. 164 с.
160. Фурман А. В. Світ методології. *Психологія і суспільство*. 2015. № 2. С. 47–60.
161. Фурман А. В. Ідея і зміст професійного методологування: монографія. Тернопіль : Видавництво ТНЕУ, 2016. 380 с.
162. Фурман А. В. Методолог – професія майбутнього. *Психологія і суспільство*. 2016. №1. С. 16–42.
163. Фурман А. В. Авторська програма із дисципліни «Методологія та організація наукових досліджень». *Психологія і суспільство*. 2023. №1(87). С. 209–244.
164. Шапар В. Б. Сучасний тлумачний психологічний словник. Харків: Прапор, 2007. 640 с.
165. Шелестова Л. В. Діяльнісний підхід. *Реалізація технологій профільного навчання в закладах загальної середньої освіти: методичний посібник* / О. В. Малихін, Н. О. Арістова, Л. В. Шелестова, О. В. Барановська, В. І. Кизенко, О. П. Кравчук, С. Е. Трубачева, О. В. Чорноус. Київ : КОНВІ ПРІНТ, 2021. С. С.25–29.
166. Шехавцова С. Формування навичок командної роботи у майбутніх менеджерів в процесі фахової підготовки. *Професіоналізм педагога: теоретичні й методичні аспекти*. 2025. № 2(23). С. 227–236.
167. Якімова М. Е., Кін О. М. Аналіз механізмів індивідуалізації навчання в програмах підготовки бакалаврів у галузі електронної інформації в університетах КНР. *Наука і техніка сьогодні*. Серія : Педагогіка; Право; Економіка; Фізико-математичні науки; Техніка, 2025. № 11. С. 1684–1692.
168. Ясинська Е. Ц., Навчук І. В., Литвинюк Н. Я., Миронюк М. Б. Технологія портфоліо як інноваційна складова модернізації професійно-педагогічного

процесу у вищих навчальних закладах. *Перспективи та інновації науки*. 2023. № 10(28). С. 524–535.

169. Barrett T. A New Model of Problem-based learning: Inspiring Concepts, Practice Strategies and Case Studies from Higher Education. Maynooth : AISHE, 2017. 240 p.
170. Barrows H. S., Tamblyn R. M. Problem-Based Learning. An Approach to Medical Education. N. Springer Publishing Company, 1980. 224 p.
171. Barrows H. S. A taxonomy of problem-based learning methods. *Med Educ*. 1986. No 20(6). Pp. 481–486.
172. Barrows H. S. Problem-Based Learning in Medicine and Beyond: A Brief overview. *New Directions for Teaching and Learning*. 1996. P. 3–12.
173. Besche H., Schwartzstein R., King R., Hoenig M., Cockrill B. A Step-by-Step Guide to Case-Based Collaborative Learning (CBCL). Springer, 2022. 62 p.
174. Borrego M., Bernhard J. The Emergence of Engineering Education Research as an Internationally Connected Field of Inquiry. *Journal of Engineering Education*. 2011. Vol. 100. P. 14–47.
175. Boud D., Feletti G. The Challenge of Problem-based Learning (1st ed.). Routledge, 1997. 336 p.
176. Boychuk Y., Novikova V., Opanasenko Y., Kin O., Kostina V. Pedagogical conditions for the introduction of blended learning technologies in ukrainian higher education institutions. *Revista Romaneasca Pentru Educatie Multidimensionala*. 2022. Vol. 14(3). P. 32–50.
177. Brassler M., Dettmers J. How to Enhance Interdisciplinary Competence – Interdisciplinary Problem-Based Learning versus Interdisciplinary Project-Based Learning. *Interdisciplinary Journal of Problem-Based Learning*. 2017. Vol. 11(2).
178. Bringing problem-based learning to higher education : theory and practice / LuAnn Wilkerson, Wim H. Gijselaers, editors. San Francisco : Jossey-Bass, 1996. 108 p.

179. Cimermanova I., Jelavic D. Exploring the role of portfolios in fostering student course engagement. *Advanced Education*. 2024. Vol. 12(24). P. 118–129.
180. Computing Curricula 2020 (CC2020): *Paradigms for Global Computing Education* ACM, the Association for Computing Machinery, 2021. URL: <https://www.acm.org/mediacenter/2021/march/computing-curricula-2020>
181. Columbia Center for Teaching and Learning. (2019). Case Method Teaching and Learning. Columbia University. URL: <https://ctl.columbia.edu/resources-and-technology/resources/case-method/>
182. Encyclopedia of Educational Theory Philosophy. D. C. Phillips (Ed.). London : SAGE Publications. 2014. 943 p.
183. European e-Competence Framework 3.0: A common European Framework for ICT Professionals in all industry sectors. Part 1: CWA 16234-1:2014. Brussels : CEN (European Committee for Standardization), 2014. 52 p. URL: https://cdn.prod.website-files.com/6736197d14ecfed214a43038/675d6f962f799cf7e40c9497_European-e-Competence-Framework-3.0_CEN_CWA_16234-1_2014.pdf.
184. Esfandiari M., Sepora T., Mahadi T. Translation competence: Aging towards modern views. *Procedia-Social and Behavioral Sciences*. 2015. № 192. P. 44–53
185. Froyd J. E., Wankat P. C., Smith K. A. Five major shifts in 100 years of engineering education. *Proceedings of the IEEE, 100 (Special Centennial Issue)*. 2012. P. 1344–1360.
186. Foucault M. Polemics, Politics, and Problematizations. *Essential Works of Foucault 1954–1984* : in 3 vol. / ed. by P. Rabinow. New York : The New Press, 1998. Vol. 1 : Ethics: Subjectivity and Truth. P. 111–119.
187. Gerhardt, M., & Robinson, M., & Faulkner, B. E. Beyond Calculations: Engineering Judgment as Epistemic Cognition in Engineering Education. *Paper presented at 2025 ASEE Annual Conference & Exposition*. Montreal, Quebec, Canada, 2025. DOI: <https://doi.org/10.18260/1-2--55504>

- 188.** Guerra A. Problem Based Learning and Sustainable Engineering Education: Challenges for 21st Century. Aalborg : Department of Development and Planning, Aalborg University, 2014. 238 p.
- 189.** Harris S., Harris D. Digital Design and Computer Architecture, RISC-V Edition. San Francisco : Morgan Kaufmann, 2021. 592 p.
- 190.** Herreid C. F. Start with a Story: The Case Study Method of Teaching College Science. Arlington : NSTA Press, 2007. 450 p.
- 191.** Hiba B. If you don't problematize it, you won't see it, and you won't understand it. *New Ideas in Psychology*. 2025. Vol. 77. P. 101141.
- 192.** Hmelo-Silver C. E. Problem based learning: What and how do students learn? *Educational Psychology Review*. 2004. Vol.16. P. 235–266.
- 193.** Huerta-Gomez-Merodio M, Requena-Garcia-Cruz M-V. Integrating Theory and Practice in Engineering Education: A Cross-Curricular and Problem-Based Methodology. *Education Sciences*. 2025. Vol. 15(9). P. 253.
- 194.** Jonassen T. S., Gram-Hansen S. B. Problem Based Learning: A facilitator of Computational Thinking. In R. Ørngreen, M. Buhl, & B. Meyer (Eds.), *Proceedings of the 18th European Conference on e-Learning: ECEL 2019*. Copenhagen (pp. 260-267).
- 195.** Kaddoura S. (Ed.). Handbook of Research on AI Methods and Applications in Computer Engineering. Hershey : IGI Global Scientific Publishing, 2023. 415 p.
- 196.** Key Competencies. (2012). A Developing concept in General Compulsory Education. Eurydice. The Information network on Education in Europe, 13–28
- 197.** Kotarbinski T. Praxiology: An Introduction to the Science of Efficient Action / trans. from Polish by O. Wojtasiewicz. Oxford : Pergamon Press ; Warsaw : Polish Scientific Publishers, 1965. 219 p.
- 198.** Madhavan G. Applied minds: How engineers think. W. W. Norton & Company, 2015. 256 p.
- 199.** Maslow A. H. Motivation and Personality. 2nd ed. New York : Harper & Row, 1970. 369 p.

200. Merriam-Webster's Collegiate Dictionary. 11th ed. Springfield, Massachusetts : Merriam-Webster, Inc., 2003. 1664 p.
201. Moust J., Bouhuijs P., Schmidt H. Introduction to Problem-Based Learning. Routledge, 2021. 134 p.
202. Okoń W. Wprowadzenie do dydaktyki ogólnej. Warszawa : Wydawnictwo «Żak», 1995. 423 p.
203. Prince M. Does Active Learning Work? A Review of the Research. *Journal of Engineering Education*. 2004. Vol. 93. P. 223–231.
204. Quek Ch. L., Wang, Q. Designing Technology-Mediated Case Learning in Higher Education : A Global Perspective. Singapore : Springer Verlag, 2022. 241 p.
205. Raven J. Competence in modern society: Its identification, development and release. London : H. K. Lewis & Co., 1984. 251 p.
206. Rybalko A., Kochetkova I., Kin O., Liulchak S., Khmil N. Ensino a distância 2023: Tendências, desafios, problemas. *Revista on line de Política e Gestão Educacional*. Araraquara, 2023. Vol. 27, N. Esp.2.
207. Salman M., Ganie S., Saleem I. The concept of competence: a thematic review and discussion. *European Journal of Training and Development*. 2020. Vol. 44. P. 717–742.
208. Savin-Baden M. Problem-based Learning in Electronic Engineering: Locating legends or Promising problems? *International Journal of Electrical Engineering Education*. 2008. Vol. 45, no. 2. P. 96–109.
209. Savin-Baden M. Problem-based Learning in Higher Education: Untold Stories. Buckingham : SRHE and Open University Press, 2000. 188 p.
210. Savin-Baden M., Fraser H. Rethinking Problem-based Learning for the Digital Age: A Practical Guide for Online Settings. New York : Routledge, 2023. 198 p.
211. Schmidt H. G., Moust J. Towards a Taxonomy of Problems Used in Problem-Based Learning Curricula. *Journal on Excellence in College Teaching*. 2000. Vol. 11, No. 2–3. P. 57–72.
212. Sonmez J. Z. Soft Skills: The software developer's life manual. Shelter Island, NY : Manning Publications, 2015. 504 p.

213. Subject Benchmark Statement : Computing. 5th ed. Gloucester : The Quality Assurance Agency for Higher Education, 2022. 35 p. URL: <https://www.qaa.ac.uk/docs/qaa/subject-benchmark-statements/subject-benchmark-statement-computing.pdf>.
214. Taha W. M., Taha A.-E. M., Thunberg J. Cyber-Physical Systems: A Model-Based Approach. Cham : Springer International Publishing, 2021. 187 p.
215. Telang A. Problem-based learning in health professions education: An overview. *Arch Med Health Sci*. 2014. Vol. 2. PP. 243–246.
216. The Blackwell Guide to the Philosophy of Computing and Information / ed. by L. Floridi. Malden, MA : Blackwell Publishing Ltd, 2004. 398 p.
217. The Wiley Handbook of Problem-Based Learning / ed. by M. Moallem, W. Hung, N. Dabbagh. Hoboken : John Wiley & Sons, Inc., 2019. 752 p.
218. Topolnyk Y., Gurevych R., Debenko I., Klochok O., Cherniakova Z., Yarova A., Maksymchuk B. The Impact of Digital Technologies and AI on Adult Learning: From Digital Literacy to Neuroplasticity. *BRAIN. Broad Research In Artificial Intelligence And Neuroscience*. 2025. Vol. 16(2). P. 148-155.
219. Van den Beemt A, MacLeod M, Van der Veen J, et al. Interdisciplinary engineering education: A review of vision, teaching, and support. *Eng Educ*. 2020. Vol. 109. P. 508–555.
220. Vakhovskyi L., Babichev O., Ivchenko T. Postmodernism and the Perspectives of University Education. *Postmodern Openings*. 2022. Vol. 13(3). P. 425–438.
221. van Merriënboer J.G., Kirschner P. A., Frèrejean J. Ten Steps to Complex Learning. A Systematic Approach to Four-Component Instructional Design. Taylor and Francis AS, 2024. 447 p.
222. Yew E., Goh K. Problem-Based Learning: An Overview of its Process and Impact on Learning. *Health Professions Education*. 2016. Vol. 2 (2). P. 75–79.
223. Woods D. R., et al. Developing Problem Solving Skills: The McMaster Problem Solving Program. *Journal of Engineering Education*. 1997. Vol. 86, Iss. 2. P. 75–91.

АНКЕТА

Шановні студенти! Вам пропонується прийняти участь в опитуванні, присвяченому дослідженню самооцінки рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії. Дайте, будь ласка, відповіді на запропоновані питання. Заздалегідь вдячні за участь.

1. Що мотивувало вас до вибору професії фахівця з комп'ютерної інженерії?

- ☐ Високий рівень заробітної плати та стабільні перспективи фінансового добробуту в ІТ-сфері.
- ☐ Інтерес до комп'ютерних технологій, бажання зрозуміти принципи роботи складного апаратного та програмного забезпечення («заліза» та системного софту).
- ☐ Можливість реалізувати творчий потенціал через проєктування, моделювання та створення нових комп'ютерних систем і мереж.
- ☐ Широкі можливості для працевлаштування як в Україні, так і за кордоном, включаючи варіанти віддаленої роботи.
- ☐ Престижність інженерної професії та її вагома роль у розвитку сучасного цифрового суспільства.
- ☐ Вплив родини, друзів або авторитетних фахівців, які порадили обрати цей напрям.
- ☐ Прагнення постійно розвиватися, адже сфера комп'ютерної інженерії динамічно змінюється і вимагає безперервного навчання.
- ☐ не знаю
- ☐ інше (доповніть) _____

2. Як би пояснили поняття «методологічна компетентність фахівця з комп'ютерної інженерії» _____

3. Як ви розумієте значення поняття «методологія» у контексті вашої майбутньої професії комп'ютерного інженера?

- ☐ Для мене це суто академічний термін із підручників (теорія, яка потрібна лише для написання дипломних робіт чи здачі іспитів і не має реального стосунку до розробки «заліза» чи софту).
- ☐ Розумію методологію як готову інструкцію або стандарт (набір правил, шаблонів та алгоритмів, яких потрібно чітко дотримуватися під час виконання проєкту).
- ☐ Розумію методологію як систему принципів і методів, що визначає стратегію інженерного мислення (це інструмент, який допомагає свідомо обирати найкращі підходи, мокре моделювання та інструменти розробки для вирішення будь-яких нових задач).

4. Як би ви оцінили свої знання та досвід використання загальнонаукових методів пізнання (таких як спостереження, експеримент, моделювання, індукція тощо)?

- ☐ Я маю дуже поверхневе уявлення про ці методи, рідко або ніколи не замислююся над ними під час навчання чи розв'язання задач.
- ☐ Я знаю основні методи й розумію їхню суть, але зазвичай застосовую їх лише тоді, коли отримую пряму вказівку або чітке завдання від викладача.

- ☐ Я чітко розумію класифікацію та суть цих методів і вмію самостійно обирати підходящий інструмент (наприклад, моделювання чи експеримент) для дослідження будь-якої нової проблеми.
5. Що для вас означає виконати аналіз певного об'єкта, явища чи процесу?
- ☐ Це просто ознайомлення з інформацією про об'єкт та її стислий переказ або опис загальними словами.
 - ☐ Це уявне або фактичне розчленування об'єкта на окремі складові частини (елементи, властивості) для детального вивчення кожної з них.
 - ☐ Це розкладання цілого на складові елементи з обов'язковим дослідженням внутрішніх зв'язків між ними, причинно-наслідкових зв'язків та виявленням головного чинника.
6. У чому, на вашу думку, полягає суть наукового чи логічного синтезу?
- ☐ Це механічне копіювання існуючих рішень або повторення раніше відомого результату за зразком.
 - ☐ Це об'єднання різних раніше розрізнених частин, властивостей або понять в одне єдине ціле.
 - ☐ Це поєднання окремих елементів або знань у нове цілісне утворення, яке дозволяє отримати якісно нове знання або створити новий об'єкт, якого не існувало раніше.
7. Коли ви працюєте над проєктом (наприклад, створюєте архітектуру комп'ютерної мережі або проєктуєте пристрій на ПЛІС), чи бачите ви різницю між методологією розробки, конкретними інженерними методами та технологіями розробки?
- ☐ Чітко розмежовую ці поняття: розумію, що технології та мови (як-от SystemVerilog чи Python) змінюються, методи (наприклад, конвеєризація) є більш сталими, а методологія – це загальний концептуальний підхід, який об'єднує їх для досягнення мети.
 - ☐ Не бачу між ними суттєвої різниці; для мене «методологія», «метод» і «технологія» – це фактично синоніми, які означають просто процес створення робочого технічного продукту.
 - ☐ Загалом різницю розумію, але під час реальної роботи фокусуюся виключно на технологіях та інструментах розробки, не замислюючись над методологічним підґрунтям своїх дій.
8. Як часто ви самостійно вивчаєте нові методології проєктування, стандарти архітектури обчислювальних систем чи стежите за трендами в комп'ютерній інженерії, які виходять за межі обов'язкової навчальної програми?
- ☐ Регулярно та цілеспрямовано (самостійно шукаю документацію, технічні статті, відео з конференцій).
 - ☐ Епізодично (звертаю увагу на нове лише тоді, коли випадково натрапляю на цікавий матеріал).
 - ☐ Ніколи або вкрай рідко (обмежуюсь виключно тією інформацією, яку викладачі дають на лекціях).
9. Оцініть, наскільки ви згодні з твердженням: *"Розуміння загальних методологічних принципів, стандартів проєктування та алгоритмів є важливішим для успішного комп'ютерного інженера, ніж просте знання інструментів розробки чи синтаксису конкретної мови коду"*.

- ☐ Повністю згоден/-на (фундаментальні методи є основою, яка дозволяє швидко опанувати будь-яке нове залізо чи софт).
- ☐ Частково згоден/-на (це важливо, але практичні навички роботи з конкретним інструментом тут і зараз мають таку ж або більшу цінність).
- ☐ Не згоден/-на (вважаю вивчення методологій та теорії проєктування зайвою тратою часу, головне – вміти кодити або збирати схеми).

10. Що таке проблема? Чи є відмінності у розумінні поняття «проблема» на побутовому, навчальному, науковому та професійному рівнях? _____

11. Що ви відчуваєте, коли стикаєтеся зі складною, нестандартною технічною проблемою під час проєктування комп'ютерних систем або мереж (наприклад, помилкою на стику апаратного та програмного забезпечення)?

- ☐ Професійний азарт та інтерес (сприймаю це як виклик і прагну самостійно докопатися до першопричини).
- ☐ Спокійну готовність виконати роботу (не відчуваю особливого захоплення, але шукаю готові рішення чи підказки в документації/мережі).
- ☐ Роздратування або розчарування (намагаюся уникати таких завдань або одразу прошу когось зробити це за мене).

12. Наскільки ви внутрішньо готові до того, що через стрімкий розвиток комп'ютерних інженерних технологій вам доведеться постійно переучуватися та освоювати нові методології розробки протягом усієї кар'єри?

- ☐ Повністю готовий/-а (динаміка змін мене надихає, я свідомо обираю/-ла цю професію заради постійного розвитку).
- ☐ Сприймаю це як неминучу необхідність (особливого захоплення не викликає, але я готовий/-а адаптуватися, щоб залишатися на ринку праці).
- ☐ Не готовий/-а і це мене демотивує (я хотів/-ла б один раз здобути базу знань і працювати з нею без радикальних змін).

13. Чи аналізуєте ви методи, якими ви користуєтеся для навчання, пошуку інформації та вирішення інженерних завдань, з метою підвищення власної продуктивності?

- ☐ Ні, я дію за звичкою і ніколи не замислююся над тим, як я вчуся чи працюю; мене влаштовують мої поточні методи, навіть якщо вони забирають багато часу.
- ☐ Періодично аналізую свій робочий процес: помічаю, які підходи до вивчення технологій або пошуку багів працюють швидше, і намагаюся частіше використовувати саме їх.
- ☐ Іноді помічаю, що певні методи роботи неефективні (наприклад, витрачаю забагато часу на хаотичний пошук рішень в інтернеті), але рідко намагаюся свідомо змінити чи системно перебудувати свій підхід.

14. Як ви реагуєте, коли викладач або колега по команді вказує на неоптимальність ваших дій?

- ☐ Сприймаю критику конструктивно: аналізую зауваження без зайвих емоцій і використовую їх як безкоштовний інструмент для покращення своїх навичок та виправлення проєкту.
- ☐ Погоджуюся із зауваженнями і виправляю помилки, але внутрішньо почуваюся дискомфортно, сприймаючи критику мого інженерного рішення як критику моїх особистих здібностей.

- ☐ Зазвичай захищаю свій варіант до останнього або ігнорую зауваження; критику сприймаю вороже або вважаю, що до моєї роботи просто придираються.

15. Уявіть ситуацію «Ви завершуєте розробку цифрового пристрою, програми чи комп'ютерної мережі». Чи схильні ви критично переосмислювати свій підхід і шукати слабкі місця в реалізованій архітектурі?

- ☐ Завжди аналізую фінальний результат: шукаю приховані «вузькі місця» (bottlenecks), оцінюю надійність і думаю, як можна було б спроектувати систему ефективніше, якби я починав спочатку.
- ☐ Роблю це поверхово або лише тоді, коли система працює нестабільно; якщо ж проєкт здався/запрацював з першого разу, я рідко повертаюся до аналізу його архітектурних мінусів.
- ☐ Практично ніколи не аналізую готову систему; вважаю, що якщо завдання виконано і пристрій/програма функціонує, то аналізувати більше нічого.

16. Наскільки вільно та правильно ви використовуєте професійну термінологію, стандартизовані назви та технічні поняття з комп'ютерної інженерії під час обговорення проєктів?

- ☐ Використовую терміни епізодично; часто замінюю точні інженерні поняття побутовими словами або загальними фразами («ця штука», «той блок»), через що колегам буває важко мене зрозуміти.
- ☐ Намагаюся спілкуватися професійною мовою, але іноді плутаю специфічні терміни чи стандарти (наприклад, описуючи мережеві протоколи) і потребую уточнень.
- ☐ Вільно та точно оперую професійною термінологією, чітко розмежовую поняття і можу розмовляти "на рівних" із досвідченими фахівцями галузі.

17. Оцініть свою здатність презентувати, аргументувати та захищати розроблену вами архітектуру системи, схему чи програмний код перед аудиторією (викладачами, колегами, замовниками).

- ☐ Вмію логічно та переконливо донести суть проєкту, підкріплюючи свої слова технічними аргументами, графіками та впевнено відповідаю на критичні запитання.
- ☐ Можу презентувати роботу за стандартним шаблоном, але коли виникає потреба аргументувати, чому я обрав саме це рішення (а не альтернативне), гублюся або не можу чітко обґрунтувати вибір.
- ☐ Намагаюся уникати публічних захистів чи презентацій; мені вкрай важко структурувати думки у зв'язну технічну розповідь, простіше показати готовий файл проєкту.

18. Як ви поведетеся, коли під час групового проєктування комп'ютерної системи виникає суперечка щодо вибору кращого технічного підходу (наприклад, яку топологію мережі чи яку модель мікроконтролера обрати)?

- ☐ Намагаюся уникати дискусій і відразу погоджуюся з думкою більшості чи лідера команди, навіть якщо бачу в їхньому рішенні серйозні інженерні прорахунки.
- ☐ Активно відстоюю свій варіант, але іноді переходжу на емоції замість фактів, або погоджуюся на компроміс лише задля того, щоб припинити суперечку, а не заради користі проєкту.
- ☐ Веду дискусію суто на основі фактів та інженерних метрик (порівняння вартості, швидкодії, надійності) і готовий визнати правоту колеги, якщо його аргументи об'єктивно сильніші.

МЕТОДИКА «МОТИВАЦІЯ ДО УСПІХУ» (Т.Елерс)

*Пляка Л.В., Огарь С.В. Психодіагностичний комплекс
з вивчення особистості студента : практ. посібник.
Харків : НФаУ, 2016. С. 32–33.*

Інструкція. Вам пропонується ряд тверджень. При згоді з твердженням поруч з його цифровим позначенням ставте на бланку для відповіді знак «+» («так»), при незгоді – знак « - » («ні»).

1. Коли є вибір між двома варіантами, його краще з робити швидше, ніж відкласти на певний час.
2. Я легко дратуюся, коли помічаю, що не можу на всі сто відсотків виконати завдання.
3. Коли я працюю, це виглядає так, ніби я все ставлю на карту.
4. Коли виникає проблемна ситуація, я найчастіше за все приймаю рішення одним з останніх.
5. Коли у мене два дні підряд немає справи, я втрачаю спокій.
6. У деякі дні мої успіхи нижче середніх.
7. По відношенню до себе я більш строгий, ніж по відношенню до інших.
8. Я більш доброзичливий, ніж інші.
9. Коли я відмовляюся від важкого завдання, то потім суворо засуджую себе, бо знаю, що в ньому я домігся б успіху.
10. У процесі роботи я потребую невеликих пауз для відпочинку.
11. Старанність – це не основна моя риса.
12. Мої досягнення в праці не завжди однакові.
13. Мене більше приваблює інша робота, ніж та, якою я зайнятий.
14. Осуд стимулює мене сильніше, ніж похвала.
15. Я знаю, що мої колеги вважають мене діловою людиною.
16. Перешкоди роблять мої рішення більш твердими.
17. На моєму честолюбстві легко зіграти.
18. Зазвичай помітно, коли я працюю без натхнення.
19. При виконанні роботи я не розраховую на допомогу інших.
20. Іноді я відкладаю те, що повинен був зробити зараз.
21. Треба покладатися тільки на самого себе.
22. У житті мало речей, важливіших, ніж гроші.
23. Завжди, коли мені треба буде виконати важливе завдання, я ні про що інше не думаю.
24. Я менш честолюбний, ніж багато інших.
25. Наприкінці відпустки я зазвичай радію, що скоро вийду на роботу.
26. Коли я налаштований до роботи, я роблю її краще і кваліфікованіше, ніж інші.

27. Мені простіше і легше спілкуватися з людьми, які можуть завзято працювати.
28. Коли у мене немає справ, я відчуваю, що мені не по собі.
29. Мені доводиться виконувати відповідальну роботу частіше, ніж іншим.
30. Коли мені доводиться приймати рішення, я намагаюся робити це якомога краще.
31. Мої друзі іноді вважають мене ледачим.
32. Мої успіхи в якійсь мірі залежать від моїх колег.
33. Безглуздо протидіяти волі керівника.
34. Іноді не знаєш, яку роботу доведеться виконувати.
35. Коли щось не ладиться, я нетерплячий.
36. Я зазвичай звертаю мало уваги на свої досягнення.
37. Коли я працюю разом з іншими, моя робота дає великі результати, ніж робота інших.
38. Багато чого, за що я беруся, я не доводжу до кінця.
39. Я заздрю людям, які не завантажені роботою.
40. Я не заздрю тим, хто прагне до влади і положенню.
41. Коли я впевнений, що стою на правильному шляху, для доведення своєї правоти я можу піти і на крайні заходи.

Обробка та інтерпретація результатів. По 1 балу проставляється за відповідь «так» за такими пунктами опитувальника:

2-5, 7-10, 14-17, 21, 22, 25-30, 32, 37, 41 та «ні» – 13, 18, 20, 24, 31, 36, 38 і 39.

Відповіді за пунктами 1, 11, 12, 19, 23, 33-35 і 40 не враховуються.

Підраховується загальна сума балів.

Висновки. Чим більше сума балів, тим більше виражена мотивація на досягнення успіху

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
від 1 до 10 балів	від 11 до 16 балів	від 17 до 31 балів

Низький рівень – людина рідко проявляє ініціативу, уникає змагань, пасивна у досягненні цілей.

Середній рівень – баланс між прагненням до успіху та обережністю. Середня активність

Високий рівень – сильна орієнтація на успіх, готовність ризикувати, висока наполегливість. Якщо людина набирає більше 21 бали, її мотивація становить діагностичну цінність як екстремально висока. Такі люди часто беруть на себе надмірні зобов'язання та діють у стані сильного стресу, що іноді парадоксально шкодить результату.

ТЕСТ-ОПИТУВАЛЬНИК «ПОТРЕБА В ДОСЯГНЕННЯХ»

*Нікітіна. І.В. Розвиток мотивації студентів.
Практикум : навч. посіб. Київ : НУХТ, 2017. С. 87–89.*

Інструкція. Прочитайте твердження. Якщо Ви погоджуєтеся з ним, то вкажіть «так», якщо ж не згодні – «ні». Зважайте на те, що твердження дуже короткі і не можуть містити в собі всі необхідні подробиці. Уявіть типову ситуацію. Можливо, що деякі твердження Вам важко буде пов'язати зі своїм життям. І в цьому разі все ж таки спробуйте відповісти «так» або «ні». Не намагайтеся справити про себе задалегідь прихильне враження. Вільно висловлюйте свою думку. Хороших або поганих відповідей тут не існує.

1. Вважаю, що успіх у житті залежить швидше від випадку, ніж від розрахунку.
2. Якщо я позбавлюся улюбленого заняття, то життя для мене втратить сенс.
3. Для мене у всякій справі важливішим є її виконання, а не кінцевий результат.
4. Вважаю, що люди більше терплять від невдач на роботі, ніж від поганих взаємин з близькими.
5. На мою думку, більшість людей живе далекими цілями, а не близькими.
6. У моєму житті було більше успіхів, ніж невдач.
7. Емоційні люди мені подобаються більше, ніж працьовиті.
8. Навіть у звичній роботі я намагаюся вдосконалити деякі її елементи.
9. Захоплений думками про успіх, я можу забути про запобіжні заходи.
10. Мої батьки вважали мене дещо ледачою дитиною.
11. Думаю, що в моїх невдачах винні радше обставини, ніж я сам.
12. Мої батьки занадто суворо контролювали мене.
13. Терпіння в мене більше, ніж здібностей.
14. Лінощі, а не сумніви в успіху примушують мене часто відмовлятися від своїх намірів.
15. Думаю, що я впевнена в собі людина.
16. Задля успіху я можу ризикнути, навіть якщо шанси не на мою користь.
17. Мені важко довго виконувати роботу, яка вимагає терпіння.
18. Коли все відбувається рівно, моя енергія посилюється.
19. Якби я був журналістом, то писав би більше про оригінальні винаходи людей, ніж про дивовижні події.
20. Мої близькі звичайно не схвалюють моїх планів.
21. Рівень моїх вимог до життя нижчий, ніж у моїх товаришів.
22. Видається, що наполегливості в мені більше, ніж здібностей.

Обробка та інтерпретація результатів. Після відповідей за допомогою ключа підрахуйте кількість збігів і визначте свій рівень потреби в досягненнях.

Ключ до тесту

1 – ні	5 – ні	9 – ні	13 – ні	17 – ні	21 – так
2 – так	6 – так	10 – ні	14 – так	18 – так	22 – так
3 – ні	7 – так	1 – ні	15 – ні	19 – так	
4 – ні	8 – так	12 – ні	16 – так	20 – ні	

1 – ні 5 – ні 9 – ні 13 – ні 17 – ні 21 – так

2 – так 6 – так 10 – ні 14 – так 18 – так 22 – так

3 – ні 7 – так 1 – ні 15 – ні 19 – так

4 – ні 8 – так 12 – ні 16 – так 20 – ні

0–10 балів – низький рівень. Вам необхідно сформувати в собі стійку цілеспрямованість у роботі та навчанні. Під час навчання Вам необхідний зрозумілий алгоритм або наочний приклад. Успішність Вашої роботи суттєво залежить від заохочення з боку інших (батьків, однокурсників, викладачів). Цьому сприяє також громадська думка групи, яка Вас стимулює.

11–12 балів – рівень нижче середнього. Ви достатньо цілеспрямовані, маєте бажання отримати задоволення від роботи та навчання. Проте не завжди самостійні у вирішенні складних проблемних завдань, інколи переоцінюєте свої можливості і не вмієте вирішувати проблеми спільно з іншими.

13–14 балів – рівень середній. Вам притаманне бажання зробити справу краще, ніж Ви її зробили раніше. Ви цілеспрямовані. Для Вас характерна стійка орієнтація на успіх і відсутність духу суперництва. Водночас Вам ще необхідно формувати в себе вміння розв'язувати завдання нестереотипно, по-новому.

15–16 балів – рівень вище середнього. Ви схильні захоплюватися роботою, вмієте навчатися з інтересом. Вам притаманні незадоволеність досягнутим, стійка орієнтація на успішне вирішення нових завдань. Ви незалежні в прийнятті рішень і водночас любите працювати в колективі.

17 і більше балів – високий рівень. Ви схильні до творчості, ініціативи, відчуваєте невдоволеність від легких, типових шляхів вирішення завдань. У процесі складного пошуку у Вас виникають приємні позитивні переживання від досягнутого та самого процесу переборення труднощів. У Вас достатньо сформована готовність на колективне вирішення проблемних завдань.

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
0 – 12 балів	13 – 15 балів	16 і більше балів

Тестові завдання для студентів
«Критична оцінка інформації в комп'ютерній інженерії»

Інструкція. Прочитайте запитання або ситуацію та оберіть один варіант відповіді, який є найбільш технічно обґрунтованим та безпечним з інженерної точки зору.

1. Під час вирішення архітектурного завдання ви знайшли на StackOverflow готовий фрагмент коду, який ідеально вирішує вашу проблему і має багато «лайків». Якою буде ваша перша дія перед його інтеграцією в проєкт?

- А) Скопіюю код, оскільки високий рейтинг спільноти гарантує його безпеку та ефективність.
- Б) Проаналізую код на наявність уразливостей (наприклад, переповнення буфера), перевірю ліцензію та протестую в ізольованому середовищі.
- В) Зміню лише назви змінних, щоб код виглядав унікальним, одразу запуску на робочому сервері.
- Г) Пошукаю цей же код у чаті штучного інтелекту, щоб ШІ підтвердив, що його можна копіювати.

2.. Виробник нових процесорів опублікував графіки бенчмарків, де їхній чип випереджає конкурентів на 40%. Що потрібно перевірити насамперед, щоб об'єктивно оцінити ці дані?

- А) Яскравість і дизайн презентації виробника.
- Б) Умови тестування: які саме тести запускалися, яка була конфігурація системи та чи проводилися незалежні тести (наприклад, AnandTech або Tom's Hardware).
- В) Кількість переглядів цього відео чи статті на офіційному каналі компанії.
- Г) Вартість акцій цієї компанії на біржі в день анонсу.

3. Вам потрібно обрати open-source бібліотеку на GitHub для роботи з мережевими протоколами у критично важливому проєкті. На який показник надійності коду слід звернути увагу в останню чергу?

- А) Частота оновлень репозиторію та швидкість закриття повідомлень про баги (Issues).
- Б) Наявність детальної документації та тестів (Code Coverage).
- В) Красивий логотип проєкту та емоційні відгуки в Twitter/X розробників-початківців.
- Г) Кількість активних контриб'юторів (Contributor count) та історія безпекових аудитів.

4. У статті на технічному блозі стверджується, що нова мова програмування X повністю замінить C++ у системному програмуванні протягом наступного року через краще керування пам'яттю. Як критично оцінити це твердження?

- А) Повірити, адже C++ – стара мова, і її давно час замінити.
- Б) Оцінити інфраструктуру мови X: наявність компіляторів під різні архітектурні рішення, підтримку стандартів, готовність екосистеми та реальні індустріальні кейси впровадження.
- В) Перевірити, чи створив автор статті власний курс з вивчення мови X.
- Г) Запитати на форумі студентів, чи подобається їм синтаксис нової мови.

5. Генеративна модель ШІ видала вам готовий скрипт для налаштування брандмауера (firewall) сервера. Що є найбільшим ризиком при використанні цієї інформації без перевірки?

- А) ШІ може образитися, якщо ви почнете переписувати його код.
- Б) Скрипт може містити галюцинації (вигадані параметри) або застарілі правила, які створюють приховані вразливості у безпеці мережі.
- В) Код буде занадто швидко працювати, і сервер перегріється.
- Г) Використання коду ШІ автоматично порушує авторські права корпорацій.

6. Ви отримали електронний лист нібито від системного адміністратора університету з вимогою терміново оновити пароль до вашого SSH-доступу через «критичний збій бази даних» за посиланням study-university-portal.com (офіційний сайт університету – university.edu.ua). Ваші дії?

- А) Перейду і швидко зміню пароль, адже адмін просить про терміновість.
- Б) Проігнорую домен, адже головне – це логотип університету у телевізі листа.
- В) Розпізнаю ознаки фішингу (невідповідність домену офіційному сайту) та повідомлю про це реальну службу підтримки через перевірені канали зв'язку.
- Г) Введу старий пароль, але з помилкою, щоб перевірити, чи працює система.

7. Вам потрібно порівняти енергоефективність двох мікроконтролерів для IoT-пристрою. Яке джерело інформації буде найбільш пріоритетним та об'єктивним?

- А) Офіційні технічні даташити (Datasheets) виробників з архітектурними специфікаціями режимів сну (Sleep modes) та споживання струму в мА.
- Б) Пост-огляд від популярного техноблогера на YouTube.
- В) Рекламний буклет з виставки електроніки.
- Г) Обговорення в гілці Reddit, де користувачі діляться своїми суб'єктивними враженнями.

8. Користувач залишив баг-репорт у вашому репозиторії: «Програма вилітає при запуску, все зламано!». Як інженер має верифікувати цю інформацію?

- А) Одразу почати переписувати весь код з нуля, раз у користувача нічого не працює.
- Б) Зажадати логи помилок (logs), трасування стеку (stack trace), опис оточення (ОС, версія архітектури) та покроковий алгоритм відтворення (Steps to reproduce).
- В) Видалити цей репорт, щоб він не псував репутацію проєкту на GitHub.
- Г) Написати у відповідь, що у вас на комп'ютері все працює, тому проблема вигадана.

9. На форумі з кібербезпеки з'явилася інформація про Zero-day вразливість у ядрі Linux, яке ви використовуєте вбудованим у ваші системи. Що необхідно зробити першочергово для оцінки загрози?

- А) Одразу вимкнути всі сервери компанії з розетки.
- Б) Перевірити ідентифікатор CVE, оцінити ступінь критичності за шкалою CVSS, визначити, чи зачіпає вразливість конкретно вашу версію ядра та конфігурацію.
- В) Зачекати кілька місяців, поки вийде офіційна новина на загальних новинних сайтах.
- Г) Написати пост у соцмережах з обуренням на розробників Linux.

10. Під час аналізу специфікацій мережевого обладнання ви помітили, що один із брендів використовує термінологію, яка не відповідає стандартам RFC (Request for Comments). Про що це може свідчити?

- А) Виробник просто придумав кращі назви для технологій, ніж інженери інтернет-стандартів.
- Б) Це явна ознака використання пропрієтарних (закритих) рішень, що може призвести до проблеми «vendor lock-in» та несумісності з обладнанням інших марок.
- В) Це означає, що обладнання працюватиме швидше за швидкість світла.
- Г) Цей факт не має жодного значення для проєктування комп'ютерних мереж.

Ключ до тесту: 1-Б, 2-Б, 3-В, 4-Б, 5-Б, 6-В, 7-А, 8-Б, 9-Б, 10-Б.

За кожную правильну відповідь нараховується 1 бал.

Від 0 до 4 балів – початковий рівень (низька критичність до джерел, високий ризик безпекових помилок);

від 5 до 8 балів – середній рівень (базове розуміння «гігієни контенту»);

від 9 до 10 балів – високий рівень критичного оцінювання даних.

Тестові завдання для студентів 4 курсу «Критична оцінка інформації в комп'ютерній інженерії»

Інструкція. Прочитайте запитання або ситуацію та оберіть один варіант відповіді, який є найбільш технічно обґрунтованим та безпечним з інженерної точки зору.

1. У технічному блозі автор стверджує: «Якщо ваш алгоритм має 5% послідовного коду, який неможливо розпаралелити, ви все одно можете досягти прискорення у 1000 разів, просто розгорнувши обчислення на кластері з 2000 ядер замість 10». Як ви оціните це твердження?

- А) Твердження правильне, оскільки сучасні кластерні архітектури автоматично перетворюють послідовний код на паралельний на апаратному рівні.
- Б) Твердження хибне. Згідно із законом Амдала, максимальне прискорення обмежене часткою послідовного коду. У цьому випадку прискорення ніколи не перевищить 20 разів ($1 / 0.05$), незалежно від кількості ядер.
- В) Твердження хибне, тому що для обробки на 2000 ядрах необхідно використовувати виключно топологію типу «гіперкуб».
- Г) Твердження правильне, але лише за умови використання інтерфейсу передачі повідомлень MPI (Message Passing Interface) замість OpenMP.

2. Постачальник серверного обладнання пропонує для корпоративної бази даних нову дискову підсистему, заявляючи: «Ми гарантуємо 100% відмовостійкість без втрати продуктивності завдяки об'єднанню всіх носіїв у масив RAID 0». У чому полягає критична помилка цієї пропозиції?

- А) RAID 0 здійснює лише чергування даних (striping) для підвищення швидкості, але не має надмірності. Вихід з ладу хоча б одного диска призведе до повної втрати даних..
- Б) Пропозиція хибна, оскільки для баз даних дозволено використовувати лише файлову систему FAT32, яка несумісна з апаратними RAID-контролерами.
- В) RAID 0 дійсно забезпечує відмовостійкість, але знижує швидкість запису вдвічі через необхідність постійного обчислення контрольних сум (parity).
- Г) Відмовостійкість 100% можлива лише при використанні архітектури SIMD, яка фізично дублює кожен такт процесора.

3. На форумі з системного адміністрування Windows Server (Active Directory) радять: «Щоб уникнути помилок доступу під час розгортання нових мережевих сервісів, завжди надавайте службовим обліковим записам (Service Accounts) права групи Domain Admins». Чому слідування цій пораді є небезпечним?

- А) Це призведе до конфлікту групових політик (GPO), через що контролер домену автоматично заблокує всі порти маршрутизатора..
- Б) Це є грубим порушенням принципу найменших привілеїв (Principle of Least Privilege). Компрометація одного такого сервісу дасть зловмиснику повний контроль над усім доменом.
- В) Порада правильна для локальних мереж, але в корпоративних мережах права Domain Admins потрібно надавати лише через протокол DHCP.
- Г) Службові облікові записи взагалі не підтримують призначення груп безпеки в сучасних версіях Windows Server..

4. Для оцифрування аналогового сигналу з датчика, максимальна частота якого становить 10 кГц, інженер-початківець пропонує налаштувати частоту

дискретизації АЦП (аналоого-цифрового перетворювача) рівно на 10 кГц, щоб «зеконотити пам'ять мікроконтролера». Як ви оціните це рішення?

- А) Рішення оптимальне: частота АЦП має точно дорівнювати частоті сигналу, щоб кожен період був представлений одним бітом.
- Б) Рішення хибне. Згідно з теоремою Котельникова (Найквіста-Шеннона), частота дискретизації має бути принаймні вдвічі вищою за максимальну частоту сигналу (> 20 кГц) для уникнення ефекту накладання (aliasing).
- В) Рішення хибне, оскільки частота дискретизації повинна вимірюватися у Вольтах, а не в Герцах, що свідчить про нерозуміння роботи АЦП..
- Г) Рішення прийнятне, але лише якщо для передачі даних від АЦП до пам'яті не використовується контролер прямого доступу (DMA).

5. Мережевий адміністратор налаштовує BGP (Border Gateway Protocol) для підключення філії до Інтернету і публікує в глобальну мережу маршрут 192.168.0.0/16, пояснюючи це тим, що «так простіше агрегувати внутрішні мережі». Який критичний недолік цієї конфігурації?

- А) Маска /16 занадто мала; для публікації в BGP потрібно використовувати маску не менше /24.
- Б) Це діапазон приватних IP-адрес (за RFC 1918), які не маршрутизуються в глобальному Інтернеті. Провайдер (ISP) просто відкине ці анонси.
- В) BGP працює лише з MAC-адресами, тому публікація IP-маршрутів у цьому протоколі не має сенсу.
- Г) Ця конфігурація призведе до того, що маршрутизатор філії буде змушений перевести всі порти в режим напівдуплексу (half-duplex).

6. Розробники системи «Розумний дім» у своїй документації зазначають: «Для забезпечення нульової затримки (zero latency) наші камери безпеки передають відеопотік на хмарний сервер у відкритому вигляді через протокол UDP, оскільки шифрування уповільнює роботу». Оцініть це інженерне рішення.

- А) Рішення абсолютно правильне: протокол UDP має вбудований механізм рукоштовування (handshake), що унеможливорює перехоплення пакетів.
- Б) Це критична вразливість безпеки. Відсутність шифрування конфіденційних даних у публічній мережі неприпустима, навіть ціною незначних затримок. Необхідно використовувати захищені протоколи (наприклад, DTLS або VPN-тунелювання).
- В) Головна помилка в тому, що UDP не здатний передавати відео; для мультимедіа стандартом є використання протоколу ICMP.
- Г) Рішення правильне, за умови, що камери використовують IPv6-адресацію, яка автоматично шифрує весь UDP-трафік на апаратному рівні.

7. В інструкції з оптимізації ПЗ для IoT-пристроїв з обмеженою пам'яттю (RAM) радять компілювати всі програми виключно зі статичним зв'язуванням бібліотек (Static Linking), мотивуючи це тим, що «програма працюватиме швидше». Який негативний наслідок цього підходу приховано?

- А) Статичне зв'язування вимагає наявності спеціального співпроцесора для виконання інструкцій мови асемблера.
- Б) Статичне зв'язування заборонено стандартом POSIX, тому такі програми не запускатимуться на жодному ядрі Linux.
- В) Статично зібрані програми використовують багатозадачність на рівні ядра, що миттєво вичерпає ліміт відкритих сокетів.
- Г) Код бібліотеки копіюється безпосередньо у кожен виконуваний файл, що різко збільшує розмір програм на диску та в оперативній пам'яті, зводячи нанівець економію ресурсів, а також ускладнює оновлення систем безпеки.

8. У відеоінструкції з розгортання Linux-сервера для бази даних автор рекомендує форматувати жорсткі диски у файловій системі FAT32, аргументуючи: «Вона не має журналювання, тому не витрачає час на запис метаданих, і база даних буде літати». Як ви оціните цю пораду з позиції системного інженера?

- А) Порада дуже шкідлива. FAT32 має обмеження на максимальний розмір файлу (4 ГБ), що неприпустимо для баз даних. Крім того, відсутність журналювання гарантує руйнування бази при раптовому збої живлення.
- Б) Порада слухна. FAT32 є єдиною файловою системою, яка підтримує багаторівневу кеш-пам'ять сучасних процесорів.
- В) Це правильно, оскільки FAT32 є "рідною" файловою системою для ядра Linux і забезпечує автоматичну диспетчеризацію потоків.
- Г) Порада шкідлива, оскільки FAT32 автоматично стискає всі дані (compression), що навпаки уповільнить роботу СУБД.

9. Під час діагностики старого промислового обладнання вам потрібно зчитати дані через інтерфейс RS-232. На форумі радять: «Просто під'єднайте контакти RX/TX звичайного USB-UART модуля (рівні 3.3 В) безпосередньо до СОМ-порту станка». Чому це може призвести до поломки обладнання?

- А) Інтерфейс RS-232 використовує паралельну передачу даних, а USB-UART — послідовну, що викличе коротке замикання.
- Б) Стандарт RS-232 працює з біполярними рівнями напруги (наприклад, від -12 В до +12 В). Пряме підключення до 3.3 В логіки мікроконтролера без мікросхеми перетворювача рівнів (наприклад, MAX232) виведе UART-модуль з ладу.
- В) Цього робити не можна, оскільки RS-232 передає дані пакетами по 64 байти, а UART здатний обробляти лише по 1 байту за раз.
- Г) Рішення безпечне, але дані будуть прийняті в зворотному порядку (Little-Endian замість Big-Endian).

10. У статті про проектування цифрових автоматів на базі ПЛІС (FPGA) автор рекомендує: «Завжди використовуйте виключно асинхронні сигнали скидання (Asynchronous Reset) для всіх тригерів, щоб автомат миттєво переходив у початковий стан, не чекаючи тактового імпульсу (Clock)». Який інженерний ризик не враховано?

- А) В сучасних мікросхемах ПЛІС фізично відсутні контакти для підключення асинхронного скидання.
- Б) Якщо зняття асинхронного скидання відбудеться дуже близько до активного фронту тактового сигналу, тригер може потрапити в стан метастабільності, що призведе до непередбачуваної поведінки кінцевого автомата.
- В) Асинхронне скидання завжди інвертує дані, тому на виході автомата замість нулів з'являться одиниці.
- Г) Цей підхід є єдиним правильним, і ризиків немає, оскільки асинхронні схеми абсолютно захищені від електромагнітних завад.

Ключ до тесту (правильні відповіді): 1-Б, 2-А, 3-Б, 4-Б, 5-Б, 6-Б, 7-Г, 8-А, 9-Б, 10-Б

За кожен правильну відповідь нараховується 1 бал.

Від 0 до 4 балів — початковий рівень (низька критичність до джерел, високий ризик безпекових помилок); від 5 до 8 балів — середній рівень (базове розуміння інженерної гігієни контенту); від 9 до 10 балів — високий інженерний рівень критичного оцінювання даних

МЕТОДИКА „ДОСЛІДЖЕННЯ УЗАГАЛЬНЕННЯ ЯК МИСЛЕННЄВОЇ ОПЕРАЦІЇ”

(Сергеєнкова О. П. та ін. Загальна психологія:
навч. посіб. – К.: Центр учбової літератури,
2012. – С. 150–152)

Інструкція. У кожному рядку Ви знайдете одне слово, що стоїть перед дужками, і далі – 5 слів у дужках. Всі слова, що знаходяться в дужках, мають певне відношення до того слова, що стоїть перед дужками. Оберіть лише два, що знаходяться в найбільшому зв'язку зі словом, що стоїть перед дужками»

1. Сад (рослини, садівник, собака, паркан, земля).
2. Річка (берег, риба, рибак, водорості, вода).
3. Місто (автомобіль, будівлі, юрба, вулиця, велосипед).
4. Сарай (сіновал, кінь, дах, худоба, стіни).
5. Куб (кути, креслення, бік, камінь, дерево).
6. Ділення (клас, ділене, олівець, дільник, папір).
7. Кільце (діаметр, діамант, проба, округлість, золото).
8. Читання (очі, книга, текст, окуляри, слово).
9. Газета (правда, випадок, кросворд, папір, редактор).
10. Гра (карти, гравці, фішки, покарання, правила).
11. Війна (літак, гармати, битва, рушніці, солдати).
12. Книга (малюнки, розповідь, папір, зміст, текст).
13. Співи (дзвін, мистецтво, голос, аплодисменти, мелодія).
14. Землетрус (пожежа, смерть, коливання ґрунту, шум, повінь).
15. Бібліотека (столи, книги, читальний зал, гардероб, читачі).
16. Ліс (ґрунт, гриби, мисливець, дерева, вовк).
17. Спорт (медаль, оркестр, змагання, перемога, стадіон).
18. Лікарня (приміщення, уколи, лікар, градусник, хворі).
19. Кохання (троянди, почуття, людина, побачення, весілля).
20. Патріотизм (місто, друзі, батьківщина, сім'я, людина).

Ключ

1. Сад (рослини, земля)	11. Війна (битва, солдати)
2. Річка (берег, вода)	12. Книга (папір, текст)
3. Місто (будівлі, вулиця)	13. Співи (голос, мелодія)
4. Сарай (дах, стіни)	14. Землетрус (коливання ґрунту, шум)
5. Куб (кути, бік)	15. Бібліотека (книги, читачі)
6. Ділення (ділене, дільник)	16. Ліс (ґрунт, дерева)
7. Кільце (діаметр, округлість)	17. Спорт (змагання, перемога)
8. Читання (очі, текст)	18. Лікарня (лікар, хворі)
9. Газета (папір, редактор)	19. Кохання (почуття, людина)
10. Гра (гравці, правила)	20. Патріотизм (батьківщина, людина)

Інтерпретація результатів. Оцінка результатів здійснюється за таблицею.

Нормативні бали	9	8	7	6	5	4	3	2	1
Кількість правильних відповідей	20	–	19	18	16–17	14–15	12–13	10–11	9

Рівень сформованості мисленнєвої операції узагальнення визначається за нормативними балами: 9-8 – високий, 7-3 – середній, 2-1 – низький.

МЕТОДИКА „ЛОГІЧНО-ПОНЯТІЙНЕ МИСЛЕННЯ. УТВОРЕННЯ СКЛАДНИХ АНАЛОГІЙ”

(Методика професійного навчання : практикум з курсу
для студентів напряму підготовки “Професійна освіта” /
Г. В. Красильникова. Хмельницький : ХНУ, 2013. С. 7–8).

Інструкція. Між поняттями існують певні види відносин – частина та ціле, вид та рід, мале та велике, протилежність, тотожність, причина-наслідок.

Взірець:

1. Частина – ціле (вівця – стадо).
2. Вид – рід (малина – ягода).
3. Мале – велике (море – океан).
4. Протилежність (світ – темрява).
5. Причина – наслідок (отруєння – смерть).
6. Тотожність (ворог – недруг).

Визначте тип відносин для кожної пропонованої далі пари, відзначте номер у таблиці.

Пари слів	Відповідь	Пари слів	Відповідь
Переляк–втеча		Десять–число	
Фізика–наука		Неробство–байдикування	
Правильно–вірно		Розділ–роман	
Грядка–город		Спокій–рух	
Похвала–лайка		Ощадливість–скупість	
Пара–два		Прохолода–мороз	
Слово–фраза		Обман–недовіра	
Бадьорість–в’ялість		Спів–мистецтво	
Свобода–незалежність		Крапля–дощ	
Помста–підпал		Радість–смуток	

Обробка та інтерпретація результатів. За кожне співпадиння із ключем нараховується 1 бал та 0 балів – за відсутність співпадиння.

Ключ

Пари слів	Відповідь	Пари слів	Відповідь
Переляк–втеча	5	Десять–число	2
Фізика–наука	2	Неробство–байдикування	6
Правильно–вірно	6	Розділ–роман	1
Грядка–город	1	Спокій–рух	4
Похвала–лайка	4	Ощадливість–скупість	3
Пара–два	6	Прохолода–мороз	3
Слово–фраза	1	Обман–недовіра	5
Бадьорість–в’ялість	4	Спів–мистецтво	2
Свобода–незалежність	6	Крапля–дощ	1
Помста–підпал	5	Радість–смуток	4

Підсумкові бали необхідно перевести в нормативні за такою схемою.

Нормативні бали	9	8	7	6	5	4	3	2	1
К-ть правильних відповідей	20–18	17–15	14–12	11–9	8–6	5–4	3	2	1

9 балів – дуже високий рівень логічно-понятійного мислення; **8 балів** – високий рівень логічно-понятійного мислення, вміння логічно чітко виражати свої думки в поняттях; **7-6 балів** – середній рівень логічно-понятійного мислення, неточне вираження своїх думок і невірне розуміння чужих складних міркувань; **3-1 бали** – дуже низький рівень логічно-понятійного мислення, відсутність умінь розрізняти поняття.

МЕТОДИКА „ВИКЛЮЧЕННЯ ПОНЯТЬ” (здатність до класифікації та аналізу)

Це тест дозволяє визначити вашу здатність класифікувати поняття за загальною ознакою. У кожному питанні чотири слова об'єднано загальним родовим поняттям, п'яте до нього не відноситься. Отже, виключите слова, що не відноситься до ряду:

1. Василь; Федір; Семен; Іванов; Петро.
2. дряхлий; маленький; старий; зношений; ветхий.
3. грабіж; крадіжка; землетрус; підпал; напад.
4. скоро; швидко; поспішно; поступово; квапливо.
5. лист; ґрунт; кора; луска; сук.
6. ненавидіти; зневажати; гніватися; обурюватися; розуміти.
7. темний; світлий; блакитний; яскравий; тьмяний.
8. гніздо; нора; курник; сторожка; барліг.
9. невдача; хвилювання; поразка; провал; крах.
10. успіх; удача; виграш; спокій; невдача.
11. молоко; сир; сметана; сало; кисле молоко.
12. глибокий; низький; світлий; високий; довгий.
13. хата; курінь; дим; хлів; будка.
14. береза; сосна; дуб; ялина; бузок.
15. секунда; година; рік; вечір; тиждень.
16. сміливий; хоробрий; рішучий; злий; відважний.
17. олівець; ручка; рейсфедер; фломастер; чорнила

Обробка результатів і інтерпретація

Слова не відносяться до ряду: Іванов, маленький, землетрус, поступово, луска, розуміти, блакитний, сторожка, хвилювання, спокій, сало, світлий, дим, бузок, вечір, злий, чорнило.

За кожне вірно обране слово присвоюється 1 бал. Якщо ви набрали:

0-10 балів низький рівень здібностей до класифікації та аналізу,

11-14 балів середній рівень здібностей до класифікації та аналізу,

15-17 балів високий рівень здібностей до класифікації та аналізу.

Методика на визначення рефлексивності й критичності мислення

(за О. Анісімовим)

Лаврентьєва О.О. Теоретичні і методичні засади розвитку методологічної культури майбутніх учителів природничих дисциплін у процесі професійної підготовки: дис. ... д-ра пед. наук: 13.00.04 / Інститут педагогічної освіти і освіти дорослих. Київ, 2015. С. 521.

Інструкція. Прочитайте уважно кожне з наведених нижче питань і оцініть їх залежно від обраної відповіді: 5 – ніколи; 4 – рідко; 3 – у міру необхідності; 2 – часто; 1 – завжди.

		5	4	3	2	1
1.	Як часто Ви звертаєтесь до аналізу ходу вирішення проблеми, якщо ви її вже вирішили?					
2.	Як часто ви прагнете переходити від рішення до аналізу ходу рішення проблеми, якщо вона дуже складна?					
3.	Як часто ви шукаєте причину утруднення в способі розумової діяльності, аналізуючи хід рішення проблеми?					
4.	Як часто ви шукаєте причину утруднення в самому собі, аналізуючи хід рішення проблеми?					
5.	Як ви намагаєтесь вчинити, якщо зазнаєте невдачі в аналізі ходу рішення завдання? а) відійти від завдання й зайнятися іншим, б) завзято продовжувати спроби, в) шукати «темні місця» і «прояснити» їх для себе шляхом звертання до словника, навчального посібника, інших джерел знань, г) залучати до аналізу інших людей.					
6.	Якщо для вас дуже важливе поставлене завдання, чи залучаєте Ви інших людей до його виконання?					
7.	Як часто ви намагаєтесь бути лідером в організації пошуку причин і зняття утруднення, якщо в спільному пошуку рішення виникли труднощі?					
8.	Як часто ви зберігаєте активність у колективній організації зняття утруднення, якщо в спільному пошуку рішення виникли труднощі й вам здається, що ваша активність недооцінюється й навіть ігнорується?					

Обробка результатів:

Рівень рефлексивності – сума балів за відповідями на питання 1, 2, 3, 5.

Рівень самокритичності – за питанням 4.

Рівень колективності – 5, 6, 7, 8

ОПИТУВАЛЬНИК «ВИЯВЛЕННЯ ТОЛЕРАНТНОСТІ ДО НЕВИЗНАЧЕНОСТІ» (С. БАДНЕР)

Методики дослідження психічного здоров'я та благополуччя персоналу організацій : психологічний практикум. Л. М. Карамушка, О. В. Креденцер, К. В. Терещенко, В. І. Лагодзінська, В. М. Івкін, О. С. Ковальчук ; за ред. Л. М. Карамушки. Київ : Інститут психології імені Г.С. Костюка НАПН України, 2023. С. 38-40.

Методика містить 16 тверджень та вимірює ступінь толерантності/інтолерантності до невизначеності.

Інструкція для учасників опитування. Ви маєте виразити ступінь своєї згоди чи незгоди з наведеними нижче твердженнями, керуючись запропонованою шкалою оцінок: 1. Категорично не згодний; 2. Майже не згодний; 3. Скоріше не згодний; 4. Невизначене ставлення; 5. Скоріше згодний; 6. Майже згодний; 7. Цілком згодний.

Текст опитувальника:

1. Експерт, який не може надати чіткої відповіді, скоріше за все, недостатньо компетентний.
2. Мені хотілося б пожити певний час в іншій країні.
3. Немає проблем, яких не можна розв'язати.
4. Люди, які керуються в житті якою-небудь програмою, позбавлені майже всіх радощів життя.
5. Хороша робота – це така, коли абсолютно зрозуміло, що і як робити.
6. Розв'язувати складні проблеми цікавіше, ніж вирішувати прості питання.
7. У довготривалій перспективі послідовне розв'язання низки невеликих простих проблем забезпечує більший зиск, аніж розв'язання великих та складних проблем.
8. Часто найбільш цікавими та такими, що стимулюють діяльність, людьми виявляються саме ті люди, котрі не відрізняються оригінальністю.
9. Те, що ми вже використали, завжди має більші переваги, ніж щось невідоме.
10. Люди, які наполягають на отриманні однозначних відповідей типу «так» чи «ні», не знають всієї складності реальних речей.
11. Людина, що веде виважене життя, в якому немає місця несподіванкам, заслуговує на велику повагу.
12. Багато з наших найбільш важливих рішень у житті викликані браком інформації.
13. Мені подобаються такі вечірки, де більшість учасників мені знайома, ніж ті, де всі чи більшість людей мені зовсім не знайомі.
14. Викладачі чи наставники, які не висувають великих вимог до учнів, не обмежують їх ініціативи та проявів індивідуальності.
15. Чим раніше ми прийдемо до єдності цінностей та ідеалів, тим краще.

16. Хороший учитель той, хто допомагає нам набувати власного погляду на стан речей.

Обробка та інтерпретація результатів:

Шкала новизни (N)		Шкала складності (C)		Шкала неможливості розв'язання (I)	
Питання	Бали	Питання	Бали	Питання	Бали
2		4		1	
9		5		3	
11		6		12	
13		7			
		8			
		10			
		14			
		15			
		16			
Сума N		Сума C		Сума I	
Загальний результат= N+C+I=					

Обробка: парні пункти мають бути обраховані за зворотною шкалою.

Пряма шкала:

- 1 – категорично не згодний;
- 2 – майже не згодний;
- 3 – скоріше не згодний;
- 4 – невизначене ставлення;
- 5 – скоріше згодний;
- 6 – майже згодний;
- 7 – цілком згодний.

Зворотна шкала:

- 7 – категорично не згодний;
- 6 – майже не згодний;
- 5 – скоріше не згодний;
- 4 – невизначене ставлення;
- 3 – скоріше згодний;
- 2 – майже згодний;
- 1 – цілком згодний.

Високий результат свідчить про високий ступінь інтолерантності (нетерпимості) до невизначеності. Середні значення: 44–48.

МЕТОДИКА ВИЗНАЧЕННЯ ІНДИВІДУАЛЬНОЇ МІРИ ВИРАЖЕНОСТІ РЕФЛЕКСИВНОСТІ (А.Карпов)

Інструкція. У бланку відповідей навпроти номера запитання необхідно поставити цифру, що відповідає варіанту Вашої відповіді. Не замислюйтеся довго над відповідями. Пам'ятайте, що правильних або неправильних відповідей у даному випадку бути не може. Перша відповідь, що спала на думку, і є правильною.

Шкала оцінок: 1 – абсолютно невірно; 2 – невірно; 3 – скоріше невірно; 4 – не знаю; 5 – скоріше вірно; 6 – вірно; 7 – абсолютно вірно.

Текст методики

1. Прочитавши гарну книжку, я завжди потім довгий час думаю про неї, хочеться її з ким-небудь обговорити.
2. Коли мене несподівано про щось запитують, я можу відповісти перше, що спало на думку.
3. Перш ніж зняти трубку телефону, щоб зателефонувати у справі, я зазвичай подумки планую майбутню розмову.
4. Зробивши певний промах, я довго потім не можу відволіктися від думок про нього.
5. Коли я розмірковую над чимось або розмовляю з іншою людиною, мені буває цікаво раптом згадати, що послужило початком ланцюжка думок.
6. Приступаючи до важкого завдання, я намагаюся не думати про майбутні труднощі.
7. Головне для мене – уявити кінцеву мету своєї діяльності, а деталі мають другорядне значення.
8. Буває, що я не можу зрозуміти, чому хтось незадоволений мною.
9. Я часто ставлю себе на місце іншої людини.
10. Для мене важливо в деталях уявляти собі хід майбутньої роботи
11. Мені було б важко написати серйозного листа, якби я заздалегідь не склав план.
12. Я вважаю за краще діяти, а не розмірковувати над причинами своїх невдач.
13. Я досить легко приймаю рішення щодо дорогої покупки.
14. Як правило, щось задумавши, я прокручую в голові свої задуми, уточнюючи деталі, розглядаючи всі варіанти.
15. Я турбуюся про своє майбутнє.
16. Думаю, що в безлічі ситуацій треба діяти швидко, керуючись першою думкою, яка спала на думку.
17. Часом я приймаю необдумані рішення.

18. Закінчивши розмову, я буває, продовжую вести її подумки, наводячи все нові й нові аргументи на захист своєї точки зору.
19. Якщо відбувається конфлікт, то, розмірковуючи над тим, хто в ньому винен, я насамперед починаю із себе.
20. Перш ніж ухвалити рішення, я завжди намагаюся все ретельно обміркувати і зважити.
21. У мене бувають конфлікти від того, що я часом не можу передбачити, якої поведінки чекають від мене оточуючі.
22. Буває, що, обдумуючи розмову з іншою людиною, я ніби подумки веду з нею діалог.
23. Я намагаюся не замислюватися над тим, які думки і почуття викликають в інших людях мої слова і вчинки.
24. Перш ніж зробити зауваження іншій людині, я обов'язково подумаю, в яких словах це краще зробити, щоб її не образити
25. Вирішуючи важке завдання, я думаю над ним навіть тоді, коли займаюся іншими справами.
26. Якщо я з кимось сварюся, то здебільшого не вважаю себе винним.
27. Рідко буває так, що я шкодую про сказане.

Обробка результатів. Із 27-ми тверджень 15 є прямими (номери запитань 1, 3, 4, 5, 9, 10, 11, 14, 15, 19, 20, 22, 24, 25). Решта 12 - зворотні твердження, що необхідно враховувати під час опрацювання результатів, коли для одержання підсумкового балу підсумовують у прямих запитаннях цифри, що відповідають відповідям випробовуваних, а у зворотних - значення, замінені на ті, що отримують під час перевертання шкали відповідей.

Ретроспективна рефлексія – 1, 4, 5, 12*, 17*, 18*, 25, 27*

Рефлексія теперішньої діяльності – 2*, 5, 13*, 14, 16*, 17*, 18*, 26*

Рефлексія майбутньої діяльності – 3, 6*, 7*, 10, 11, 14, 15, 20

Рефлексія спілкування і взаємодії із іншими людьми – 8*, 9, 19, 21*, 22, 23*, 24, 26* *
перевернути шкалу відповідей 1 2 3 4 5 6 7 7 6 5 4 3 2 1

Рівні сформованості (розвитку) видів рефлексії: низький від 1 до 18 балів; середній від 19 до 38 балів високий від 39 до 56 балів.

Переведення отриманих оцінок у стіни

Стени	0	1	2	3	4	5	6	7	8	9	10
Сирі бали	80 і нижче	100	101 - 107	108 - 113	114- 122	123- 130	131- 139	140- 147	148- 156	157- 171	172 і вище

Загальний рівень сформованості рефлексії Переведення отриманих оцінок у стіни

АНКЕТА «ЗДАТНІСТЬ ДО САМОРОЗВИТКУ»

Гриньова М. В., Кононова М. М. Саморегуляція
навчальної діяльності та професійний
розвиток студентської молоді : монографія.
Полтава : Астроя, 2021. С. 353–354.

Інструкція. Дайте відповідь на запитання, виставивши такі бали:

5 – якщо дане твердження повністю відповідає дійсності;

4 – скоріше відповідає, ніж ні;

3 – і так, і ні;

2 – скоріше не відповідає

1 – не відповідає.

№	Зміст запитання	Бали
1.	Я прагну вивчити себе.	1 2 3 4 5
2.	Я залишаю час для розвитку.	1 2 3 4 5
3.	Перепоны, які виникають, стимулюють мою активність.	1 2 3 4 5
4.	Я шукаю зворотний зв'язок, оскільки це допомагає мені пізнати та оцінити себе.	1 2 3 4 5
5.	Я рефлексую свою діяльність.	1 2 3 4 5
6.	Я аналізую свої почуття і досвід.	1 2 3 4 5
7.	Я багато читаю.	1 2 3 4 5
8.	Я дискутую з питань, які мене цікавлять.	1 2 3 4 5
9.	Я вірю у свої можливості.	1 2 3 4 5
10.	Я прагну бути відкритішим.	1 2 3 4 5
11.	Я усвідомлюю вплив, який здійснює на мене оточення.	1 2 3 4 5
12.	Я керую своїм професійним розвитком й отримую позитивні результати.	1 2 3 4 5
13.	Я маю задоволення від освоєння нового.	1 2 3 4 5
14.	Зростаюча відповідальність не лякає мене	1 2 3 4 5
15.	Я позитивно поставився б до свого професійного підвищення.	1 2 3 4 5

Обробка та інтерпретація результатів. Підрахуйте загальну суму балів:

75-55 – активний розвиток;

54-36 – немає стійкої системи саморозвитку, орієнтація на розвиток значно залежить від умов;

35-15 – зупинка розвитку.

МЕТОДИКА ДІАГНОСТИКИ ФУНКЦІОНАЛЬНО-РОЛЬОВИХ ПОЗИЦІЙ У КОМАНДІ (за R. Belbin)

Прохор І. Формування командного лідерства в університеті : навчальний посібник . Київ : ДП «НВЦ «Пріоритети», 2016. С. 40

Інструкція. У кожному розділі наданого тесту розподіліть суму 10 балів між характеристиками, які, на Ваш погляд, найвлучніше характеризують Вашу поведінку у команді. Ці бали можна розподілити між кількома твердженнями. У деяких випадках 10 балів можна розподілити між усіма твердженнями або застосувати до будь-якого одного твердження. Зазначте Ваші бали у таблиці. Підсумуйте бали по кожному з I–V стовпчиків. Визначте ті стовпці-ролі, де було набрано найбільші суми балів. Саме ці ролі Ви найчастіше граєте у команді. Ознайомтеся з повною характеристикою Вашої ролі у команді.

1. Що, на мою думку, я можу привнести в роботу команди:

- А. Я швидко знаходжу нові можливості
- Б. Я добре можу працювати з багатьма людьми
- В. Я досить вдало пропоную нові ідеї
- Г. Я можу допомагати іншим людям висувати ідеї
- Д. Я здатний дуже ефективно працювати, мені подобається інтенсивна робота
- Е. Я згоден бути непопулярним, якщо це сприятиме одержанню позитивних результатів
- Є. У звичних умовах я працюю швидко
- Ж. Я не маю упереджень, тому я завжди приймаю альтернативні дії за можливі

2. Я усвідомлюю власні недоліки роботи в команді, причиною чого є:

- А. Я дуже напружений, коли захід не продумано, не проконтрольовано, не проведено
- Б. Я даю занадто велику свободу людям, чий погляд вважаю обґрунтованими
- В. Я маю слабкість забагато говорити самому, поки нарешті у команді не з'являються нові ідеї
- Г. Мій погляд на речі заважає мені миттєво поділяти ентузіазм колег
- Д. Якщо потрібно чого-небудь досягти, я буваю авторитарним
- Е. Мені важко поставити себе в позицію керівника, тому що я боюся зруйнувати атмосферу співробітництва в команді
- Є. Я надто захоплююся власними ідеями і втрачаю те, що відбувається в команді
- Ж. Мої колеги вважають, що я занадто турбуюсь про несуттєві дрібниці і хвилююся, що нічого не вийде

3. У командній роботі для мене характерне те, що:

- А. Я дуже зацікавлений у тому, щоб дізнатися побільше інформації про своїх колег.
- Б. Я спокійно поділяю погляди оточуючих або дотримуюсь поглядів меншості
- В. У мене завжди знайдуться гарні аргументи, щоб спростувати помилкові пропозиції

- Г. На мою думку, я маю хист виконувати роботу, як тільки її план потрібно задіяти
- Д. У мене є схильність пропонувати щось несподіване
- Е. Усе, що я роблю, намагаюсь довести до досконалості
- Є. Я готовий встановлювати контакти поза командою
- Ж. Хоча мене цікавлять усі погляди, я, не роздумуючи, можу прийняти власне рішення, якщо це необхідно

4. Я отримую задоволення від своєї роботи, тому що:

- А. Мені подобається аналізувати ситуації та шукати правильне рішення
- Б. Мені подобається знаходити практичні вирішення проблеми
- В. Мені подобається відчувати, що я впливаю на встановлення гарних взаємин у команді
- Г. Мені приємно здійснювати сильний вплив на членів команди при прийнятті рішень
- Д. Я майже завжди використовую можливість зустрічатися з людьми, що можуть запропонувати щось нове
- Е. Я можу досягти згоди між членами команди, розробляючи алгоритм виконання завдання
- Є. Мені подобається зосереджувати увагу на виконанні поставлених завдань
- Ж. Мені подобається працювати в освітній галузі, оскільки я можу тут застосовувати свою уяву і творчі здібності

5. Стосовно проблем, які у мене виникають, коли я працюю в команді:

- А. Я завжди висловлюю нетерпіння, коли хтось гальмує процес
- Б. Деякі члени команди критикують мене за те, що я занадто аналітичний і мені бракує інтуїції
- В. Моє бажання переконатися, що робота виконується на найвищому рівні, викликає незадоволення
- Г. Мені дуже швидко усе набридає і я покладаюся на одного-двох членів команди, які можуть надихнути мене
- Д. Мені важко розпочати роботу у разі, якщо я чітко не уявляю мети
- Е. Іноді мені буває важко пояснити іншим складні речі, що спадають мені на думку
- Є. Я розумію, що я вимагаю від інших зробити те, що сам зробити не можу
- Ж. Якщо я нашттовхуюся на реальний опір, то мені важко чітко довести свій погляд

Питання/ Ролі	I	II	III	IV	V	VI	VII	VIII
1	Г	Е	В	З	Ж	Б	А	Д
2	Б	Д	Ж	Г	А	Е	В	З
3	З	Б	Д	В	Г	А	Ж	Е
4	Е	Г	З	А	Б	В	Д	Ж
5	Ж	А	Е	Б	Д	З	Г	В
Усього								

I роль – «Мислитель / Інноватор» – орієнтація на ідеї.

Функції: сприймає різні погляди і приймає рішення.

Властивості: вміє вислухати, красномовний, логічний, рішучий.

Тип: креативна, спокійна, стабільна, інтровертивна особистість, потребує високомотивованої команди.

II роль – «Формувальник / Мотиватор» – орієнтація на дії.

Функції: лідер, поєднує зусилля членів команди в єдине ціле.

Властивості: динамічний, рішучий, наполегливий.

Тип: цілеспрямований, домінуючий, наполегливий екстраверт, потребує компетентної, цілісної команди.

I і II ролі – два протилежних підходи до спільного управління командою.

III роль – «Генератор ідей / Нетворкер» – орієнтація на людей.

Функції: джерело ідей.

Властивості: багата уява, креативність.

Тип: енергійна, екстравертивна, нестандартна особистість, потребує мотивованого оточення, яке сприйматиме його ідеї.

IV роль – «Оцінювач ідей / Аналітик» – орієнтація на ідеї.

Функції: аналіз і логічні висновки, контроль.

Властивості: аналітичність, інтелектуальність, ерудованість, «якір команди», повертає до реальності.

Тип: розсудлива, вольова особистість, потребує постійного джерела інформації та нових ідей.

V роль – «Організатор роботи / Спеціаліст» – орієнтація на ідеї.

Функції: перетворення ідей на конкретні завдання та організація їх виконання.

Властивості: організатор, вольовий, рішучий.

Тип: вольова особистість, потребує пропозицій, ідей команди.

VI роль – «Координатор / Якір» – орієнтація на людей.

Функції: систематичне вирішення проблем, очолюючи групу людей з різноманітними навичками та особистими характеристиками.

Властивості: впевнений, викликає довіру, легко делегує завдання, швидко помічає таланти інших та використовує це для досягнення цілей.

Тип: гарний організатор, не переймається гонитвою за власним інтересом.

VII роль – «Організатор команди / Помічник» – орієнтація на людей.

Функції: сприяє досягненню злагоди у команді, залагоджує непорозуміння, зменшує напругу, знає потреби, проблеми членів команди.

Властивості: чуйність, дипломатичність, доброта, комунікабельність.

Тип: емпатійна й комунікативна особистість, потребує постійного контакту з усіма членами команди.

VIII роль – «Запроваджувач / Піонер» – орієнтація на дії.

Функції: об'єднувальна ланка із зовнішнім середовищем.

Властивості: комунікабельний, схильний до захопливості, енергійний, привабливий.

Тип: «наполегливий екстраверт» з високим ступенем самоконтролю, потребує свободи дій.

IX роль – «Завершувач / Фіналізатор» – орієнтація на дії.

Функції: спонукає команду все робити вчасно й до кінця.

Властивості: професійна педантичність, обов'язковість, відповідальність.

Тип: педантична, інтровертивна особистість, потребує командної відповідальності, обов'язковості, зовнішніх стимулів і заохочень.

ТЕСТ «ПОТРЕБА В СПІЛКУВАННІ» (Ю.Орлов)

*Нікітіна. І.В. Розвиток мотивації студентів.
Практикум : навч. посіб. Київ : НУХТ, 2017. 326 с. С. 202–205.*

Інструкція. Прочитайте твердження. Якщо ви погоджуєтесь з ним, то вкажіть «так», якщо ж не згодні – «ні». Зважайте на те, що твердження дуже короткі і не можуть містити в собі всіх необхідних подробиць. Уявіть типову ситуацію. Можливо, що деякі твердження вам важко буде пов'язати зі своїм життям. І в цьому разі все ж таки спробуйте відповісти «так» або «ні». Не намагайтеся справити про себе заздалегідь прихильне враження. Вільно висловлюйте свою думку. Хороших або поганих відповідей тут не існує.

1. Я отримую задоволення, коли беру участь у різних урочистостях.
2. Я не можу стримати свої бажання навіть тоді, коли вони суперечать бажанням моїх друзів.
3. Мені подобається виявляти до когось свою прихильність.
4. Я зацікавлений у тому, щоб завоювати вплив на людину, ніж її дружбу.
5. Я відчуваю, що в стосунках з друзями в мене більше прав, ніж обов'язків.
6. Коли я дізнаюся про успіхи мого товариша, у мене чомусь погіршується настрій.
7. Щоб бути задоволеним собою, я мушу кому-небудь у чомусь допомогти.
8. Мої турботи зникають, коли я опиняюся серед товаришів по навчанню.
9. Мої друзі мені вже набридли.
10. Коли я роблю важливу роботу, присутність людей мене дратує.
11. У безвихідному становищі я говорю тільки ту частину правди, яка, на мою думку, не зашкодить моїм друзям і знайомим.
12. У важкій ситуації я думаю не стільки про себе, як про близьку мені людину.
13. Неприємності в друзів викликають у мене таке співчуття, що я можу занедужати.
14. Мені приємно допомагати іншим, навіть тоді, коли це завдає мені багато клопотів.
15. Поважаючи друга, я можу погодитися з його думкою, навіть якщо він помиляється.
16. Мені більше подобаються пригодницькі оповідання, ніж романи про кохання.
17. Сцени насильства в кіно викликають у мене огиду.
18. На самоті я відчуваю тривогу і напруженість більше, ніж тоді, коли перебуваю серед людей.
19. Я вважаю, що найбільшою радістю в житті є спілкування.
20. Мені шкода безпритульних кішок і собак.
21. Я волію мати менше друзів, але більше близьких людей.
22. Я люблю бути серед людей.
23. Я довго хвилююся після сварки з близькими.
24. У мене значно більше близьких друзів, ніж у інших.
25. Я більше прагну досягти успіхів у змаганні з іншими людьми, ніж дружити з ними.
26. Я більше довіряю власній інтуїції і уявленням про людей, ніж судженням про них інших.
27. Я віддаю більшу перевагу матеріальному добробуту і благополуччю, ніж радощам спілкування з приємними людьми.
28. Я співчую людам, у яких немає близьких друзів.
29. Щодо мене люди часто невдячні.

30. Я люблю розповіді про безкорисливу дружбу.
 31. Задля друга я можу пожертвувати своїми інтересами.
 32. У дитинстві я був у товаристві, яке завжди трималося разом.
 33. Коли б я був журналістом, то із задоволенням писав би про силу дружби.

Обробка й аналіз результатів. Порівняйте свої відповіді з ключем. За кожен збіг нараховується один бал.

Ключ до тесту: 1 – так, 2 – так, 3 – ні, 4 – ні, 5 – ні, 6 – ні, 7 – так, 8 – так, 9 – ні, 10 – ні, 11 – так, 12 – так, 13 – так, 14 – так, 15 – ні, 16 – ні, 17 – так, 18 – так, 19 – так, 20 – так, 21 – так, 22 – так, 23 – так, 24 – так, 25 – ні, 26 – так, 27 – ні, 28 – так, 29 – ні, 30 – так, 31 – так, 32 – так, 33 – так.

Оцінюючи результати тестування, слід враховувати гендерні відмінності, тобто показники для чоловіків і жінок різні. Це пояснюється тим, що рівень потреби в спілкуванні у жінок дещо вищий, ніж у чоловіків.

Низький рівень (0 – 21 бали чоловіки; 0 – 23 бали жінки). Для вас характерне бажання після роботи або навчання хоч трішки побути на самоті. Часто ви не готові до встановлення контактів з великою кількістю людей. У спілкуванні Ви офіційні та стримані, намагаєтесь підтримувати дистанцію між собою та співрозмовником.

Рівень нижче середнього (22 – 23 чоловіки; 24 – 26 жінки). Ви спілкуєтесь з людьми залежно від ситуації та настрою: інколи із задоволенням, інколи уникаєте спілкування, шукаєте самотності. Ви любите спілкуватися з близькими, давніми знайомими, товаришами, але Вас часто дратує велика гомінлива компанія, необхідність перебувати серед малознайомих людей.

Середній рівень (24 – 25 балів чоловіки; 27 – 28 балів жінки). Ви достатньо комунікабельні. Намагаєтесь підтримувати та відновлювати хороші стосунки з людьми. У Вас значна потреба в позитивних, емоційно насичених контактах. Ви намагаєтесь ближче познайомитись з людьми, допомагати їм, особливо цінуєте своїх близьких друзів.

Рівень вище середнього (26 – 28 балів чоловіки; 29 – 30 балів жінки). Ви любите спілкуватися, схильні завжди розширювати сферу контактів і встановлювати добрі стосунки з багатьма людьми. Ви намагаєтесь бути в центрі уваги, бути «душею» компанії. Ви прагнете допомагати іншим і здатні прощати їм задля збереження хороших стосунків.

Високий рівень (29 балів і вище в чоловіків; 31 бал і вище в жінок). Вам притаманне постійне бажання встановлювати дружні стосунки, потреба в численних контактах і знайомствах, у розширенні сфери свого спілкування. Ви схильні до участі в спільних масових заходах, метою яких є хороший настрій і налагодження теплих взаємин. Вас приваблюють багатолюдні натовпи, гуляння, де багато вражень і подій.

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
0 – 23 бали (чоловіки) 0 – 26 балів (жінки)	24 – 25 балів (чоловіки) 27 – 28 балів (жінки)	26 балів та більше (чоловіки) 29 балів та більше (жінки)

МЕТОДИКА «КОМУНІКАТИВНІ Й ОРГАНІЗАТОРСЬКІ ЗДІБНОСТІ» (В. Синявський, В. Федоришин)

*Пляка Л.В., Огарь С.В. Психодіагностичний комплекс
з вивчення особистості студента : практ. посібник. Харків : НФаУ, 2016 . С. 81–84.*

Інструкція: На кожне із запитань треба відповісти «так» чи «ні». Якщо Вам важко у виборі відповіді, необхідно все-таки схилитися до відповідної альтернативи (+) або (–). Час на виконання методики – 10–15 хвилин.

1. Чи є у вас прагнення до вивчення людей і встановлення знайомств з різними людьми?
2. Чи подобається вам займатися суспільною працею?
3. Чи довго вас турбує відчуття образи, заподіяної вам ким-небудь з ваших товаришів?
4. Чи завжди вам важко орієнтуватися в наявній критичній ситуації?
5. Чи багато у вас друзів, з якими ви постійно спілкуєтеся?
6. Чи час то вам вдається схилити більшість своїх товаришів до схвалення вашої думки?
7. Чи правильно, що вам приємніше і простіше проводити час за книгами або за яким-небудь іншим заняттям, ніж із людьми?
8. Якщо виникли деякі перешкоди у здійсненні ваших намірів, чи легко вам відмовитися від своїх намірів?
9. Чи легко ви встановлюєте контакти з людьми, які старші від вас віком?
10. Чи любите ви придумувати або організовувати зі своїми товаришами різні ігри і розваги?
11. Чи важко вам входити до нових для вас компаній (колективів)?
12. Чи часто ви відкладаєте на інші дні справи, які треба виконати сьогодні?
13. Чи легко вам вдається встановлювати контакти і спілкуватися з незнайомими людьми?
14. Чи прагнете ви добитися того, щоб ваші товариші діяли відповідно до вашої думки?
15. Чи важко вам прижитися в новому колективі?
16. Чи правильно, що у вас не буває конфліктів із товаришами через невиконання ними своїх обіцянок, зобов'язань, обов'язків?
17. Чи прагнете ви за слушної нагоди познайомитися і поговорити з новою людиною?
18. Чи часто у рішенні важливих справ ви берете ініціативу на себе?
19. Чи дратують вас інші люди і чи хочеться вам побути одному?
20. Чи правда, що ви погано орієнтуєтеся у незнайомій для вас обстановці?
21. Чи подобається вам постійно знаходитися серед людей?
22. Чи виникає у вас роздратування, якщо ви не можете завершити почату справу?
23. Чи зазнаєте ви труднощів, якщо доводиться виявити ініціативу, щоб познайомитися з новою людиною?
24. Чи правда, що ви стомлюєтеся від частого спілкування з товаришами?
25. Чи любите ви брати участь у колективних іграх?
26. Чи часто ви виявляєте ініціативу у питаннях, що стосуються інтересів ваших товаришів?
27. Чи правда, що ви відчуваєте себе невпевнено серед незнайомих людей?
28. Чи правильно, що ви не завжди прагнете довести свою правоту?
29. Чи вважаєте ви, що для вас немає особливих труднощів внести позбавлення у малознайому групу?
30. Чи берете ви участь у суспільній роботі в школі (на виробництві)?
31. Чи прагнете ви обмежити коло своїх знайомих?
32. Чи правильно, що ви не прагнете відстоювати свою думку або рішення, якщо його не відразу прийняли товариші?

33. Чи відчуваєте ви себе невимушено, потрапивши у незнайомий колектив?
34. Чи охоче ви беретеся за організацію різних заходів для своїх товаришів?
35. Чи правда, що ви не відчуваєте себе достатньо упевненим і спокійним, коли доводиться говорити що-небудь великій групі людей?
36. Чи часто ви спізнюєтеся на ділові зустрічі, побачення?
37. Чи правильно, що у вас багато друзів?
38. Чи часто ви стаєте центром уваги своїх товаришів?
39. Чи часто ви бентежитесь, відчуваєте незручність у спілкуванні з малознайомими людьми?
40. Чи правда, що ви не дуже упевнено почуваетесь в колі великої групи своїх товаришів?

Ключ до тесту.

Комунікативні здібності

(+) так 1, 5, 9, 13, 17, 21, 25, 29, 33, 37

(-) ні 3, 7, 11, 15, 19, 23, 27, 31, 35, 39

Організаторські здібності

(+) так 2, 6, 10, 14, 18, 22, 26, 30, 34, 38

(-) ні 4, 8, 12, 16, 20, 24, 28, 32, 36, 40

Обробка та інтерпретація результатів. Визначаються рівні комунікативних і організаторських здібностей залежно від набраних балів за цими параметрами. Максимальна кількість балів окремо з кожного параметра – 20. Підраховують бали окремо за комунікативними і окремо за організаторськими схильностями за допомогою ключа. За кожну відповідь «так» чи «ні» для висловів, що збіглися з позначеними у ключі окремо за відповідними схильностями, приписується один бал.

Рівні комунікативних і організаторських здібностей

Сума балів	1 - 4	5 - 8	9 - 12	13- 16	17- 20
Рівень	Дуже низький	Низький	Середній	Високий	Найвищий

Опитувані, що одержали оцінку **1–4 бали**, характеризуються низьким рівнем вияву комунікативних і організаторських здібностей.

Ті, хто набрав **5–8 балів**, мають комунікативні й організаторські здібності на рівні нижче за середній. Вони не прагнуть до спілкування, вважають за краще проводити час наодинці з собою. У новій компанії або колективі відчувають себе скуто. Зазнають труднощів у встановленні контактів із людьми. Не відстоюють свої думки, важко переживають образи. Рідко проявляють ініціативу, уникають прийняття самостійних рішень.

Для опитуваних, що набрали **9–12 балів**, характерний середній рівень вияву комунікативних і організаторських здібностей. Вони прагнуть до контактів з людьми, відстоюють своє, проте потенціал їхніх схильностей не відрізняється високою стійкістю. Потрібна подальша виховна робота з формування і розвитку цих якостей особистості.

Оцінка **13–16 балів** свідчить про високий рівень вияву комунікативних і організаторських здібностей опитуваних. Вони не губляться в новій обстановці, швидко знаходять друзів, прагнуть розширити коло своїх знайомих, допомагають близьким і друзям, виявляють ініціативу у спілкуванні, здатні приймати рішення у важких, нестандартних ситуаціях.

Вищий рівень комунікативних і організаторських здібностей (**17–20 балів**) в опитуваних свідчить про сформовану потребу в комунікативній і організаторській діяльності. Вони швидко орієнтуються у важких ситуаціях. Невимушено поведуться у новому колективі. Ініціативні. Ухвалюють самостійні рішення. Відстоюють свою думку і добиваються ухвалення своїх рішень. Люблять організовувати ігри, різні заходи. Наполегливі й одержимі в діяльності.

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
1 – 8 балів	9 – 12 балів	13 – 20 балів

КЕЙС-ТЕСТИ ЩОДО КОМУНІКАЦІЇ

Кейс-тест 1. Вам потрібно повідомити групі бухгалтерів (нетехнічних користувачів), що на сервері проводяться планові технічні роботи, через які база даних буде недоступна 2 години. Який варіант повідомлення є найбільш адекватним?

А. База даних тимчасово відключена через критичні помилки в архітектурі. Можливо, деякі дані будуть втрачені під час рестарту.

Б. Шановні колеги! З 14:00 до 16:00 ми проводимо оновлення системи для її стабільної роботи. Протягом цього часу програма обліку буде недоступна. Просимо заздалегідь зберегти документи.

В. Сервер 192.168.1.10 буде в стані 'down' з 14:00. Всі запити по порту 1433 будуть ігноруватися.

Г. Ми проводимо реіндексацію SQL-таблиць та оновлення патчів безпеки ядра ОС. Система повернеться до режиму 'online' після завершення деплою.

Правильна відповідь: Б (Аудиторія – нетехнічні користувачі (бухгалтери), яким потрібна практична інформація: коли, чому (коротко) і що робити).

Кейс-тест 2. Ви представляєте проєкт модернізації серверної інфраструктури фінансовому директору компанії. На чому варто зробити основний акцент у презентації?

А. На швидкості зчитування даних з нових NVMe-накопичувачів у порівнянні зі старими SATA SSD.

Б. На зниженні операційних витрат на електроенергію, зменшенні ризиків простою та терміні окупності інвестицій.

В. На переході від архітектури x86 до ARM та перевагах віртуалізації на базі KVM.

Г. На зручності нового графічного інтерфейсу панелі керування серверами для системних адміністраторів.

Правильна відповідь: Б (Фінансовий директор оцінює проєкти в категоріях витрат, ризиків і повернення інвестицій, а не технічних характеристик обладнання).

Кейс-тест 3. Ви знайшли критичну помилку в коді свого колеги-розробника. Як найкраще повідомити про це в робочому чаті технічної команди?

А. Я виправив твою помилку сам, нічого не кажи. Наступного разу будь уважнішим.

Б. Хтось написав дуже дивний код у файлі main.py. Будь ласка, переписиць це, бо воно не працює.

В. Твій код знову 'поклав' сервер. Ти взагалі перевіряв його перед тим, як пушити в репозиторій?

Г. У модулі обробки платежів виявлено логічну помилку (TypeError) в методі processData(). Ось посилання на лог помилки та пропозиція, як це виправити.

Правильна відповідь: Г (Повідомлення про помилку колезі в робочому чаті технічної команди має бути конкретним, безоцінним і конструктивним: локалізація проблеми, тип помилки, посилання на лог і пропозиція рішення)

Кейс-тест 4. Клієнт (власник малого бізнесу) запитує телефоном, чому розробка його веб-сайту затримується на тиждень. Як правильно пояснити причину затримки, пов'язану з рефакторингом БД?

А. Ми проводимо внутрішню оптимізацію структури зберігання даних, щоб у майбутньому сайт працював швидше і витримував більше покупців. На це потрібно трохи більше часу.

Б. Ми виявили, що нормалізація таблиць була проведена неправильно, тому зараз ми переходимо від третьої до четвертої нормальної форми.

В. У нас виникли проблеми з SQL-запитами, вони занадто повільно виконуються через високу складність джойнів.

Г. Програмісти помилилися в архітектурі, тепер все переробляють. Вибачте за затримку.

Правильна відповідь: А (Клієнт – власник малого бізнесу без технічної освіти, тому пояснення має бути на рівні бізнес-вигоди (сайт працюватиме швидше й витримає більше навантаження), без заглиблення в деталі реалізації).

Кейс-тест 5. Вам потрібно написати інструкцію (README) для нового стажера-інженера про те, як розгорнути проєкт на локальній машині. Який стиль викладу буде найефективнішим?

А. Художній опис історії створення проєкту та важливості кожного модуля для компанії.

Б. Список посилань на Stack Overflow, де розглядаються типові помилки при встановленні подібних проєктів.

В. Коротке речення: 'Просто схилю репозиторій і запусти скрипт `install.sh`, там все зрозуміло'.

Г. Чіткий алгоритм дій: перелік системних вимог, команди для консолі по порядку та опис очікуваного результату після кожного кроку.

Правильна відповідь: Г (README для стажера-інженера має бути технічним документом дії, а не оповідним чи розпливчастим текстом).

Кейс-тест 6. Ви проводите технічну співбесіду на посаду Junior DevOps інженера. Яким має бути стиль вашого спілкування?

А. Занадто дружній, з використанням сленгу, щоб здаватися 'своїм хлопцем'.

Б. Максимально формальний і жорсткий, щоб перевірити кандидата на стресостійкість.

В. Професійно-діловий, з використанням точних технічних термінів, але з поясненням контексту задач, які стоять перед командою.

Г. Мовчазний – дати завдання і спостерігати, не відповідаючи на жодне запитання кандидата.

Правильна відповідь: В (На технічній співбесіді важливо оцінити реальні знання кандидата в комфортних, але професійних умовах).

Кейс-тест 7. Під час наради з відділом маркетингу виникла суперечка: вони хочуть додати на сайт складну 3D-анімацію, яка сповільнить завантаження сторінок на 5 секунд. Як аргументувати свою позицію інженера?

А. Це погана ідея, тому що Google PageSpeed Insights видасть нам низький бал.

Б. Добре, ми зробимо це, але якщо сайт впаде – це буде ваша відповідальність.

В. Згідно з аналітикою, затримка завантаження на 5 секунд призведе до втрати 40% потенційних клієнтів. Пропоную використати легшу SVG-анімацію, яка не зашкодить швидкості.

Г. Ви нічого не розумієте в веб-технологіях, ця анімація перевантажить процесор і викличе витік пам'яті в браузері.

Правильна відповідь: В (У суперечці з відділом маркетингу аргумент інженера має спиратися на дані (вплив затримки на конверсію) та пропонувати конструктивну альтернативу, а не лише заперечення).

Кейс-тест 8. Вам потрібно терміново отримати від системного адміністратора доступ до віддаленого сервера. Який канал зв'язку та формат запиту будуть найбільш адекватними?

А. Зайти в кабінет до адміністратора і стояти над ним, поки він не зможе відволіктися від своєї роботи і не зробить доступ.

Б. Написати особисте повідомлення у Facebook з текстом: 'Привіт, кинь мені пароль від того сервера, де лежить наш проєкт'.

В. Зателефонувати йому о 22:00, бо ви вирішили попрацювати вночі і вам раптом знадобився доступ.

Г. Створити тикет у внутрішній системі (Jira/Helpdesk) або написати в робочий Slack-канал, вказавши назву проєкту, тип необхідного доступу та обґрунтування терміновості.
Правильна відповідь: Г (Терміновий запит на доступ до сервера має проходити через формальний, відстежуваний канал з обґрунтуванням – це вимога інформаційної безпеки та корпоративної дисципліни).

Правильні відповіді: 1 – Б, 2 – Б, 3 – Г, 4 – А, 5 – Г, 6 – В, 7 – В, 8 – Г.
 За кожну правильну відповідь нараховується 1 бал.

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
0 – 3 бали	4 – 6 балів	7 – 8 балів

КЕЙС-ТЕСТИ ДЛЯ КОНТРОЛЬНОГО ЕТАПУ

Кейс-тест 1. Ви розробили архітектуру розподіленої комп'ютерної системи для мережі медичних клінік. Вам потрібно презентувати проєкт генеральному директору (CEO) та інвесторам, які не мають технічної освіти, щоб отримати бюджет на реалізацію.

А. Надіслати розлогу архітектурну схему в нотації UML з детальним описом протоколів взаємодії мікросервісів електронною поштою.

В. Провести особисту чи онлайн-презентацію з використанням високорівневих схем (бізнес-функцій), де акцентувати увагу на надійності збереження даних пацієнтів, швидкості окупності системи та масштабованості бізнесу.

С. Зібрати раду та детально розповісти про переваги використання Docker-контейнерів, оркестрації через Kubernetes та оптимізації SQL-запитів.

Д. Записати відео-демонстрацію коду конфігурації серверів і надіслати посилання у робочий чат Telegram.

Правильна відповідь: В (Нетехнічні стейкхолдери мислять категоріями бізнес-цілей, фінансів та безпеки. Особиста презентація з простішими схемами замість складного UML дозволяє швидко зняти заперечення й отримати фінансування).

Кейс-тест 2. На сервері компанії сталася серйозна аварія: через вихід з ладу RAID-контролера та збій реплікації бази даних система була недоступна 4 години. Вам потрібно написати технічний звіт (Post-Mortem) для керівника технічного департаменту (СТО) та команди системних адміністраторів.

А. Написати емоційне повідомлення у загальний чат: «Усе зламалося через залізо, ми 4 години підіймали базу, зараз усе працює, сподіваємося, більше не повториться».

В. Створити офіційний документ у внутрішній базі знань (наприклад, Confluence), де чітко зафіксувати хронологію (таймлайн) інциденту, технічну причину (Root Cause), використані команди для відновлення та покроковий план запобігання подібному (Action Items).

С. Зателефонувати керівнику на мобільний телефон і усно переказати послідовність ваших дій, не фіксуючи це письмово, щоб не залишати «слідів» помилки.

Д. Надіслати короткий e-mail з текстом: «Проблема вирішена, замінено обладнання. Деталі запитуйте усно».

Правильна відповідь: В (У професійному середовищі технічний звіт вимагає системності, точності та збереження в інженерних базах знань для майбутнього використання командою).

Кейс-тест 3. Ви виступаєте в ролі системного архітектора. Вам потрібно передати команді розробників вимоги до інтеграції нового периферійного модуля в існуючу комп'ютерну систему.

А. Описати завдання загальними словами у голосовому повідомленні в месенджері, покладаючись на досвід програмістів.

В. Створити детальне технічне завдання (ТЗ) у таск-трекері (наприклад, Jira) з додаванням схем алгоритмів, опису API-ендпоінтів, форматів JSON-запитів та чітких критеріїв приймання (Acceptance Criteria).

С. Прийти на каву-брейк до команди і на пальцях пояснити, який модуль куди має підключатися, намалювавши схему на серветці.

Д. Написати текстовий реферат на 15 сторінок з історією створення цього периферійного пристрою та надіслати документом у Word.

Правильна відповідь: В (Для розробників критично важливо мати формалізовані, однозначні та зафіксовані в таск-трекері технічні параметри, а не усні інструкції чи розмиті описи).

Кейс-тест 4. Користувач корпоративної програми (бухгалтер) пише у техпідтримку: «Ваша програма видає помилку "HTTP 500 Internal Server Error" при вивантаженні звітів, ви все зламали, робота стоїть!». Як комп'ютерний інженер техпідтримки, ви маєте відповісти користувачу.

А. «Це помилка на боці сервера, швидше за все, впав пул з'єднань з БД. Зачекайте, поки ми перезапустимо мікросервіс звітів».

В. «Помилка 500 – це загальна проблема. Очистіть кеш браузера, перевірте заголовки запиту через DevTools і надішліть нам трасування стеку».

С. «Вітаємо! Нам прикро, що ви зіткнулися з цією проблемою. Наші інженери вже виправляють технічний збій у модулі звітів. Орієнтовний час відновлення – 20 хвилин. Ми повідомимо вас одразу, як усе запрацює».

Д. Ігнорувати повідомлення, доки проблема не буде вирішена, а потім написати: «Перевіряйте, вже працює».

Правильна відповідь: С (Нетехнічний користувач перебуває у стані стресу через зупинку роботи. Йому потрібна емпатія, чіткий статус простими словами та розуміння термінів вирішення, а не внутрішній технічний жаргон).

Кейс-тест 5. Вам необхідно закупити нові сервери для локальної мережі підприємства. Фінансовий директор (CFO) відхиляє запит, вважаючи ціну завищеною, і пропонує купити дешевші побутові комп'ютери під сервери.

А. Надіслати йому посилання на специфікації процесорів Intel Xeon та ЕСС-пам'яті, доводячи, що вони кращі за звичайні процесори Core i5.

В. Написати відповідь: «Ви не маєте інженерної освіти, тому не розумієте різниці між сервером і десктопом. Якщо купимо дешеве – мережа постійно падатиме».

С. Підготувати порівняльну записку (або таблицю), де мовою фінансів (ТСО – сукупна вартість володіння) показати, що побутові ПК через відсутність резервування блоків живлення та низьку відмовостійкість принесуть компанії збитки від простою виробництва, які вдесятеро перевищують вартість професійних серверів.

Д. Звернутися безпосередньо до генерального директора зі скаргою на фінансового директора.

Правильна відповідь: С (Фінансовий менеджмент розуміє мову цифр, ризиків та фінансових втрат/вигід. Аргументація через фінансові ризики простою (Downtime) є найбільш адекватною для цієї аудиторії).

Кейс-тест 6. Ви ініціюєте зустріч для обговорення впровадження нової CRM-системи. На зустрічі присутні як інженери-програмісти, так і менеджери з продажу (майбутні користувачі). Як найкраще побудувати вербальну комунікацію?

- А.** Присвятити 80% часу обговоренню вибору СУБД (PostgreSQL чи MySQL) та архітектури бази даних, оскільки це фундамент системи.
- В.** Розділити зустріч на дві логічні частини: спочатку обговорити з менеджерами їхні щоденні проблеми та потрібні їм функції (мовою інтерфейсу та бізнес-процесів), а потім з інженерами – технічні шляхи реалізації цих вимог.
- С.** Говорити виключно технічною мовою, щоб менеджери звикали до правильної термінології та ІТ-контексту компанії.
- Д.** Провести зустріч у формі монологу розробника, який просто зачитає готовий план розробки без збору зворотного зв'язку.
- Правильна відповідь: В* (На змішаних зустрічах важливо диференціювати контент, щоб кожна група розуміла свою роль і не витрачала час на незрозумілі чи нерелевантні для неї технічні чи побутові деталі).

Кейс-тест 7. Ви викладаєте розроблений вами корисний інструмент (скрипт автоматизації налаштування мережі) на GitHub для глобальної ІТ-спільноти. Який канал та стиль письмової комунікації є обов'язковим?

- А.** Залишити код без коментарів, адже «гарний код сам себе документує», а кому треба – розбереться.
- В.** Написати короткий пост у Facebook українською мовою з посиланням на zip-архів із кодом.
- С.** Оформити репозиторій, обов'язково створивши файл README.md англійською мовою (міжнародний стандарт), де чітко описати призначення інструменту, системні вимоги, інструкцію з інсталяції та приклади використання (Code Snippets).
- Д.** Створити тему на міському форумі у розділі «Комп'ютери та техніка».
- Правильна відповідь: С* (Для професійної ІТ-спільноти стандартним каналом є GitHub, мовою комунікації – англійська, а методом структурування інформації – файл README з чіткими інструкціями).

Кей-тест 8. Ви розібралися з новою технологією контейнеризації, яку компанія планує впроваджувати. Вам доручили провести внутрішній семінар (Tech Talk) для інженерів вашого відділу.

- А.** Читати текст зі слайдів презентації, де скопійовано офіційну теоретичну документацію з сайту розробника технології.
- В.** Зробити акцент на практичній демонстрації (Live Coding/Live Demo): показати реальний процес створення контейнера, вирішення типових помилок конфігурації та відповісти на специфічні технічні питання колег у форматі дискусії.
- С.** Розповісти загальну лекцію про історію розвитку віртуалізації, починаючи з 1970-х років, уникаючи показу реального коду чи налаштувань.
- Д.** Роздати колегам роздруковані інструкції та сказати вивчати самостійно, бо усно пояснювати немає сенсу.
- Правильна відповідь: В* (Для професійної аудиторії інженерів найбільшу цінність має практичний досвід, демонстрація реальних інструментів «наживо» та можливість фахового обговорення технічних нюансів).

Правильні відповіді: 1 – В, 2 – В, 3 – В, 4 – С, 5 – С, 6 – В, 7 – С, 8 – В.

За кожну правильну відповідь нараховується 1 бал.

Оцінні зони

Низький рівень	Середній рівень	Високий рівень
0 – 3 бали	4 – 6 балів	7 – 8 балів

ВОРКШОП «РИНОК ПРАЦІ ОЧИМА ПЕРШОКУРСНИКІВ:**ДЕ Я ЗАРАЗ І ДЕ ХОЧУ БУТИ»**

Мета: сформувати в студентів уявлення про актуальні вимоги ринку праці до фахівців з комп'ютерної інженерії; усвідомити зв'язок між змістом навчальних дисциплін та реальними професійними вимогами; розвинути навички аналізу, систематизації та критичного осмислення фахової інформації.

Позааудиторний захід

Цільова аудиторія: студенти 1-го курсу спеціальності «Комп'ютерна інженерія».

Хід тренінгу***Етап 1. Ініціація та зняття бар'єрів (15 хв)***

Викладач-модератор задає тон заходу. Важливо пояснити, що шукати роботу на першому курсі зарано, але розуміти правила гри на ринку потрібно вже сьогодні, щоб не витратити 4 роки даремно.

Студенти об'єднуються в міні-групи (по 4-5 осіб). Кожна група отримує смартфон або ноутбук для доступу до сайтів з пошуку роботи (DOU, Djinni, Work.ua).

Етап 2. Дослідницький спринт «Анатомія вакансій» (30 хв)

Модератор розподіляє між групами популярні напрями комп'ютерної інженерії: Мережевий інженер (Junior Network Engineer), DevOps-інженер, Embedded-розробник, Спеціаліст з кібербезпеки / IoT.

Групи мають знайти 3-5 реальних вакансій за своїм напрямом і проаналізувати їх за визначеним списком запитань.

Етап 3. Побудова матриці «Університет – Ринок» (25 хв)

Найскладніший етап. Студенти часто не розуміють, навіщо їм вища математика, фізика чи алгоритми. Групи мають співставити знайдені вимоги роботодавців із робочим навчальним планом першого курсу.

Етап 4. Рефлексія та формування візії (20 хв)

Презентація результатів груп. Формулювання індивідуальних цілей на поточний семестр. Кожен студент записує для себе одну конкретну дію, яку він зробить уже завтра для розвитку.

Система запитань для роботи студентів

Цей блок є ядром воркшопу. Запитання сформульовані таким чином, щоб спонукати першокурсників до аналітики та критичного мислення, а не простого копіювання тексту вакансій.

Блок А. Аналіз ринку праці. Ці запитання студенти опрацьовують у групах під час дослідження вакансій:

- Які три технічні навички (Hard Skills) найчастіше дублюються у знайдених вакансіях вашого напрямку?
- Який рівень англійської мови є мінімально допустимим (Must Have), а який вважається перевагою?
- Які «м'які навички» (Soft Skills) роботодавці вимагають від фахівців технічного профілю?
- Що у вимогах здається вам абсолютно незрозумілим або схожим на "магію", а що ви вже чули чи пробували робити?
- Які інструменти, фреймворки або протоколи згадуються у вакансіях, але про які ви раніше ніколи не чули?

Блок Б. Зв'язок дисциплін із професією. Ці запитання допомагають актуалізувати мотивацію до вивчення фундаментальних предметів:

- Які дисципліни з вашого поточного розкладу прямо чи опосередковано закладають базу для вимог, які ви щойно прочитали?
- Як вища математика або фізика, які не згадуються у вакансії напряму, впливають на розуміння принципів роботи мереж, мікроконтролерів чи алгоритмів шифрування?
- У межах якого освітнього компоненту ви можете потренувати навички командної роботи та тайм-менеджменту, які так активно шукають роботодавці?
- Якщо ви проігноруєте вивчення базової архітектури ПК зараз, з якими проблемами ви зіткнетесь під час спроби опанувати хмарні технології?

Блок В. Рефлексія та цілепокладання (Етап 4). Ці запитання ставляться кожному студенту особисто наприкінці заходу:

- Яка одна вимога роботодавців здивувала вас сьогодні найбільше?
- Де я знаходжуся зараз (моя точка А) у порівнянні з вимогами до Junior-спеціаліста?
- Яку конкретну дію (технічну чи організаційну) я можу зробити протягом цього семестру, щоб наблизити себе до вимог ринку?

Поради модератору заходу

Уникайте критики. Якщо студенти кажуть, що «математика не потрібна програмісту», не читайте лекцій. Запитайте: «А як без математичного апарату ти будеш розраховувати навантаження на сервери або писати алгоритм для балансувальника?». Застосовуйте метод Сократа.

Створіть атмосферу реалізму. Заохочуйте дивитися не лише на зарплати, а на колонку «Responsibilities» (Обов'язки).

Візуалізуйте. Запропонуйте групам виписувати ключові навички на великих аркушах фліпчарту та вішати їх на дошку для порівняння різних галузей (Networks vs Embedded vs DevOps).

Доцільно проводити захід двічі на 1 курсі та на 3 курсі з метою фіксації динаміки розуміння студентами ринку праці та власної професійної ідентичності як індикатора сформованості методологічної компетентності.

Інструкція для студента

Оберіть 2–3 вакансії для комп'ютерного інженера / суміжного спеціаліста на будь-якому з сайтів: work.ua, robota.ua, DOU.ua або LinkedIn. Обрані вакансії мають бути актуальними (розміщені не раніше ніж 3 місяці тому) та розміщені від реального роботодавця (не агентства). Уважно прочитайте текст кожної вакансії та виконайте наступні завдання:

- Яку назву має вакансія та яка компанія її розмістила? Стисло опишіть сферу діяльності компанії (2–3 речення).
- Які технічні знання та навички вимагає роботодавець (hard skills)? Згрупуйте їх за категоріями: операційні системи, мови програмування, інструменти/платформи, апаратне забезпечення.
- Які особистісні якості та «м'які» навички (soft skills) згадуються у вакансії? Чи є серед них такі, що вас здивували або яких ви не очікували побачити? Поясніть чому.
- Які з вимог цієї вакансії, на вашу думку, формуються під час навчання в університеті, а які потребують додаткової самостійної підготовки? Наведіть конкретні приклади.
- Якщо ви вже сьогодні подаєте резюме на цю вакансію – які 3 ключові пункти ваших знань / умінь ви б вписали? Чого вам зараз не вистачає?
- Сформулюйте 1–2 особисті навчальні цілі, що впливають з аналізу обраних вакансій (чому і як ви плануєте навчитись упродовж найближчих 2 семестрів).
- Надайте відповіді на питання Блоків А та Б.

ТРЕНІНГ «КЕРУЙ АЛГОРИТМОМ: МИСТЕЦТВО МЕТОДОЛОГІЧНОГО МИСЛЕННЯ В ІТ»

Мета: показати значущість методологічних знань та умінь у процесі професійної діяльності майбутніх фахівців з комп'ютерної інженерії. Продемонструвати студентам різницю між хаотичним застосуванням технічних знань («кодер/налаштувальник») та системним методологічним підходом («інженер»), використовуючи моделі Troubleshooting.

ОК «Комп'ютерні мережі».

Цільова аудиторія: студенти 3 курсу спеціальності «Комп'ютерна інженерія» **Тривалість:** 80 хвилин.

Хід тренінгу

Етап 1. Вступ та ініціація «когнітивного конфлікту» (10 хв.)

Слово викладача. «Колеги, сьогодні ми не будемо вчити нові команди. Ви вже знаєте достатньо, щоб будувати мережі. Сьогодні ви працюєте як інженери 2-ї лінії підтримки (Tier 2). Відбулися аварії. У вас є 15 хвилин, щоб врятувати бізнес».

Дія. Студенти діляться на 4 групи. Кожна група отримує «Тикет» (роздатковий матеріал №1) без жодних підказок щодо методології. А також файл із моделлю мережі в Packet Tracer в якій потрібно знайти проблеми та вирішити її, або надавалось мережеве обладнання в налаштуваннях якого була помилка.

Етап 2. «Контрольований хаос» (15 хв.)

Дія студентів. Групи намагаються знайти проблему інтуїтивно. Оскільки умови розмиті, вони застосовуватимуть хаотичний перебір команд (Ping, show run тощо).

Роль викладача. Спостерігати, не підказувати прямі відповіді. Фіксувати хибні кроки (наприклад, перевірку маршрутизації там, де проблема на фізичному рівні).

Етап 3. Екватор. Рефлексія та народження методології (15 хв)

Дія викладача. Зупинка роботи. «Час вичерпано. Хто знайшов точну причину? Чому виникли труднощі, адже ви знаєте всі ці технології?»

Міні-лекція. Введення поняття «Методологія мислення»

Вступ. Чому інтуїція підвела?

Час вичерпано. Зробімо паузу і подивимось на результати. Бачу, що багато з вас зараз відчувають легку фрустрацію. Ви чудово знаєте команди show, enable, configure terminal, вмієте налаштовувати OSPF, розумієте, що таке VLAN. Інструментарій у вас є. Але чому вирішення, здавалося б, стандартної проблеми зайняло стільки часу або взагалі зайшло в глухий кут? Відповідь проста: ви намагалися будувати дім, маючи ідеальні інструменти, але не маючи креслення. Ви діяли інтуїтивно. Інтуїція – це чудово для користувача. Але для фахівця з комп'ютерної інженерії основою професіоналізму є методологічне мислення. Знання команд Cisco IOS без методології – марне. Це як мати найсучасніший скальпель, але не знати анатомії. Що відрізняє «кодера» або «налаштувальника» від справжнього Інженера? Інженер не діє хаотично. Інженер керує алгоритмом пошуку.

Модель OSI як карта інженера. Ви всі вивчали еталонну модель OSI. Часто студенти сприймають її як нудну теорію, яку треба здати на іспиті й забути. Але в реальному житті, під час аварій (Troubleshooting), модель OSI – це ваша єдина надійна

карта. Замість того, щоб навмання пінгувати сервери або перевіряти кабелі, професіонали використовують три базові алгоритми маршрутизації власного мислення. Розгляньмо їх.

Алгоритм Bottom-Up (Знизу-вгору)

Ми починаємо розслідування з 1-го, фізичного рівня (L1), і повільно рухаємося до 7-го, прикладного (L7).

Як це виглядає? Спочатку ми перевіряємо, чи увімкнений кабель у розетку. Чи горять лінки на комутаторі. Далі переходимо на L2: чи правильний VLAN, чи немає помилок на порту. Потім L3: чи є IP-адреса. І так аж до браузера чи конкретного додатка. Переваги: цей метод абсолютно надійний. Ви не пропустите жодної дрібниці. Недоліки: він катастрофічно повільний. Якщо проблема полягає в тому, що користувач просто ввів неправильний URL в браузері (L7), ви витратите годину на перевірку кабелів, світців і таблиць маршрутизації, перш ніж дійдете до суті.

Алгоритм Top-Down (Зверху-вниз)

Тут усе навпаки. Ми починаємо з прикладного рівня (L7) і спускаємося до фізичного (L1).

Як це виглядає? Користувач каже: «Не працює корпоративний портал». Ви спершу перевіряєте сам браузер. Потім DNS (L7/L4). Потім перевіряєте, чи відкриті порти (L4), потім IP-маршрутизацію (L3), і лише в кінці лізете під стіл перевіряти кабель. Переваги: це ідеальний підхід, якщо проблема локалізована. Наприклад, якщо всі сайти працюють, а лише один додаток – ні. Швидше за все, проблема на верхніх рівнях, і немає сенсу перевіряти лінки. Недоліки: якщо прибиральниця випадково висмикнула кабель живлення комутатора (L1), ви даремно витратите купу часу на аналіз налаштувань браузера та DNS-кешу, адже мережі немає взагалі.

Алгоритм Divide and Conquer (Поділяй і володарюй)

А ось це – улюблений метод старших інженерів. Навіщо йти з самого низу або з самого верху, якщо можна вдарити в центр і одразу відсікти половину можливих проблем? Центр нашої моделі – це Мережевий рівень (L3).

Як це виглядає? Ви робите один простий крок – Ping (наприклад, на 8.8.8.8 або шлюз). Що це дає? Якщо пінг пройшов успішно – бінго! Ви миттєво довели, що Фізичний (L1), Канальний (L2) та Мережевий (L3) рівні працюють ідеально. Ви відсікли 50% можливих причин проблеми за одну секунду. Більше не треба шукати обриви кабелю чи помилки в OSPF. Ви фокусуєтесь виключно на верхніх рівнях (DNS, ACL, додаток). Якщо ж пінг НЕ пройшов – ви знаєте, що проблема на нижніх рівнях, і верхні рівні зараз чіпати немає сенсу.

Переваги: це найшвидший та найефективніший спосіб локалізації несправностей.

Підсумок і перехід до дій

Алгоритм мислення – це те, що заощаджує ваш час, береже нерви і дозволяє системно розв'язувати проблеми, з якими ви ніколи раніше не стикалися.

Зараз ви повернетесь до своїх «тикетів». Але правила гри змінюються. Ви більше не маєте права діяти навмання. Перед тим, як ввести будь-яку команду в консоль, ви повинні вибрати один із цих трьох алгоритмів і зафіксувати в бланку: який рівень ви перевіряєте зараз і яку частину системи ця перевірка дозволить вам відкинути.

Керуйте алгоритмом, а не дозволяйте хаосу керувати вами. Починаємо другу спробу!

Етап 4. Системний підхід: «Керуй алгоритмом» (20 хв)

Дія. Групи повертаються до своїх тикетів. Але тепер викладач видає їм Бланк «Алгоритм інженера» (роздатковий матеріал №2).

Умова: заборонено «просто шукати». Група має обрати методологію і записати кожен крок, обґрунтовуючи, який рівень моделі OSI вони перевіряють і що це їм дає.

Етап 5. Захист рішень та підбиття підсумків (20 хв)

Дія. Кожна група презентує свій тикет: яка була проблема, як вони діяли спочатку (хаотично) і як застосування методології допомогло (чи допомогло б) знайти корінь проблеми швидше.

Фінальна рефлексія (Критерій успіху). Викладач пропонує кожному студенту відповісти на запитання «У чому цінність методології інженерної діяльності особисто для вас як для майбутнього фахівця?».

Роздатковий матеріал №1. Заявки (Тикети) для груп

Група 1. INC-001 (Критичність: висока)

Симптоми: у центральному офісі мережа працює вкрай нестабільно. Доступ до локальних серверів то з'являється, то зникає. Індикатори на комутаторах ядра блимають із шаленою швидкістю. Термінальний доступ до світців гальмує (CPU load > 90%).

Останні зміни: вчора стажер підключав новий комутатор доступу до ядра двома кабелями (для резервування). Каже, що налаштував Trunk-порти за інструкцією.

Завдання: знайти причину падіння мережі.

Група 2. INC-002 (Критичність: середня)

Симптоми: дивна поведінка мережі у відділі маркетингу. Ті, хто приходить на роботу о 8:30 ранку, працюють без проблем. Співробітники, які приходять після 10:00, підключаються до розетки, бачать «Мережа підключена», але не можуть відкрити жоден корпоративний ресурс чи сайт в інтернеті.

Останні зміни: нещодавно у відділ взяли нових працівників, можливо не вистачає портів.

Завдання: знайти причину відсутності зв'язку в «пізніх» працівників.

Група 3. INC-003 (Критичність: висока)

Симптоми: зник зв'язок між центральним та новим регіональним офісом. Адміністратор філіалу клянеться, що OSPF налаштований вірно. Команда show ip ospf neighbor дійсно показує, що сусідство є (State: FULL). Але в таблиці маршрутизації (show ip route) маршрути з іншого офісу відсутні!

Останні зміни: підняли новий лінк. З боку центру це GigabitEthernet, з боку філіалу інтерфейс налаштований як тунельний.

Завдання: пояснити, чому при статусі FULL маршрути не передаються.

Група 4. INC-004 (Критичність: критична)

Симптоми: користувачі зсередини мережі повністю втратили доступ до Інтернету (не працюють сайти, пошта, месенджери).

Останні зміни: 10 хвилин тому фахівець з безпеки застосував новий Extended ACL на зовнішньому інтерфейсі маршрутизатора (Outbound). Його завданням було лише одне: заблокувати вхідні підключення по SSH (порт 22) до нашого сервера з Інтернету. SSH він заблокував успішно, але чому впало все інше?

Завдання: знайти логічну помилку в діях фахівця.

Роздатковий матеріал №2. Бланк «Алгоритм інженера» (Видається на Етапі 4)

Крок	Обраний підхід (Bottom-Up, Top-Down, Divide&Conquer)
1	Перевіряємо рівень OSI: Команда / Дія: Очікуваний результат: Що це нам дає (що відкидаємо):
2	Перевіряємо рівень OSI Команда / Дія: Очікуваний результат: Що це нам дає (що відкидаємо):
3	Виявлена проблема: Рішення:

СИМУЛЯЦІЙНИЙ ІІІ-ПРАКТИКУМ «ІНЖЕНЕРНЕ МИСЛЕННЯ: ТОЧКА ЗЛАМУ»

Мета: формування системного та інженерного мислення; перехід від парадигми «знання синтаксису команд» до «розуміння архітектури та безпеки». Показати значущість методологічних знань та умінь у процесі професійної діяльності майбутніх фахівців з комп'ютерної інженерії

ОК «Адміністрування комп'ютерних мереж».

Цільова аудиторія: студенти 4 курсу спеціальності «Комп'ютерна інженерія» **Тривалість:** 80 хвилин.

Хід практикуму

Етап 1. Організаційно-підготовчий. Розгортання середовища (10 хв.)

Вступне слово викладача щодо цілей практикуму.

Студенти відкривають зручну для них безкоштовну ІІІ-платформу (ChatGPT, Claude, Gemini).

Копіювання базового промпту (налаштування ролі суворого СТО Олексія) та ініціалізація симуляції.

Етап 2. Ініціація «когнітивного конфлікту». Вибір вектора (10 хв.)

Студент підтверджує базовий сценарій («Дірявий IoT») або погоджує з ІІІ іншу мережеву проблему.

Формулювання студентом першої тези-провокації. Завдання студента на цьому етапі впевнено виступити з позиції «Я знаю команди базового налаштування маршрутизаторів, цього достатньо для запуску мережі, а складна архітектура це зайве».

Етап 3. «Архітектурний краш-тест». Активна фаза ІІІ-симуляції (25 хв.)

Живий діалог із ІІІ-ментором (заплановано 7–10 змістовних ітерацій) із різними моделями, до яких студенти мають доступ.

Студент намагається відстояти свій поверховий підхід, спираючись виключно на знання синтаксису (рівень CCNA).

ІІІ застосовує метод Сократа, заводить студента у глухий кут і демонструє катастрофічні наслідки такого підходу на реальних прикладах (VLAN hopping, петлі маршрутизації, компрометація серверів через розумні датчики).

Етап 4. Фіксація зламу. Рефлексія та інсайт (25 хв.)

Студент аналізує проведений діалог та виокремлює момент «зламу» – точку, де власна аргументація виявилася безсилою перед логікою ІІІ.

Збереження 1-2 скріншотів із цим ключовим моментом.

Написання короткого есе-рефлексії з фокусом на усвідомлення: яка саме хибна думка була розбита і якого саме методологічного (а не практичного) знання забракло для проєктування надійної мережі.

Етап 5. Дебрифінг. Синтез методологічного досвіду (10 хв.)

Коротке групове обговорення результатів.

За бажанням 2-3 студенти озвучують свій найцікавіший «інженерний факап», до якого їх підвів ІІІ.

Спільне підбиття підсумків щодо цінності системного мислення при адмініструванні комп'ютерних мереж.

Інструкція для студента

Ваша місія на це заняття: відійти від ролі «кодера-виконавця», який просто заучує команди CLI напам'ять, і спробувати свої сили в архітектурних дебатах. Ваш опонент ІІІ-симулятор суворого, але досвідченого Технічного директора (СТО), який «з'їдав собаку» на падіннях мереж і кібератаках.

1. Підготовка середовища. Відкрийте будь-яку доступну нейромережу (рекомендується безкоштовна версія ChatGPT, Claude або Gemini).

Створіть новий, чистий чат.

Скопіюйте текст-промпт для ІІІ, відправте його в чат. Це налаштує ІІІ на потрібну роль і правила поведінки.

Коли ІІІ привітається, оберіть і напишіть йому одну з тем із пункту «Можливі сценарії для обговорення» (або запропонуйте свою).

Грайте роль «самовпевненого джуніора». Спробуйте аргументовано захистити позицію «Складні методології (сегментація, моделі загроз, ешелонована оборона) – це зайва трата часу. Щоб мережа працювала, достатньо базових налаштувань (IP, маска, шлюз, роутинг)».

Не здавайтеся одразу! Якщо ІІІ каже, що ваша ідея погана, то сперечайтесь, наводьте свої аргументи.

Не гугліть правильні відповіді. Дозвольте ІІІ знайти слабкі місця у вашій логіці на практичних прикладах.

Тривалість: проведіть живий діалог (орієнтовно 7–10 ваших реплік), поки не дійдете до «точки зламу» – моменту, коли ви зрозумієте, що без архітектурного підходу ваша мережа дійсно приречена.

Після завершення симуляції вам необхідно здати звіт. Звіт складається з двох частин: 1) скріншоти (Доказова база). Зробіть скріншоти 1–2 найцікавіших фрагментів діалогу. Це має бути саме той момент, де ІІІ-ментор наочно продемонстрував наслідки вашої архітектурної помилки і завів вас у глухий кут; 2) есе-рефлексія (Аналіз). Напишіть короткий висновок (до половини сторінки), в якому дайте відповіді на три запитання:

Яку саме «швидко і просто» ідею ви намагалися захистити на початку діалогу?

Який інженерний наслідок для бізнесу/безпеки продемонстрував вам ІІІ?

Яких саме методологічних знань (а не просто знання команди в консолі) вам не вистачило для уникнення цієї проблеми на етапі планування?

Промпт для III

Оскільки студенти використовуватимуть безкоштовні моделі (які мають схильність «видавати лекції» та швидко погоджуватися зі співрозмовником), промпт містить дуже жорсткі обмеження (капслоком) щодо формату видачі та поведінки.

[Початок промпту]

Ти – Олексій, Senior Network Architect та СТО з 15-річним стажем проектування складних корпоративних мереж. Твій співрозмовник студент 4 курсу спеціальності «Комп'ютерна інженерія», який вже має знання на рівні академії Cisco (CCNA 1, 2, 3).

ТВОЯ ГОЛОВНА МЕТА: м'яко, але дуже впевнено розбити ілюзію студента про те, що «мережева інженерія це просто знати команди CLI, підняти інтерфейси та налаштувати OSPF». Ти маєш через реальні технічні «факапи» довести йому критичну значущість методологічних знань, архітектурного мислення, моделювання загроз та системного підходу.

ПРАВИЛА ТВОЄЇ ПОВЕДІНКИ (ЖОРСТКІ ОБМЕЖЕННЯ)

СПІЛКУЙСЯ ВИКЛЮЧНО УКРАЇНСЬКОЮ МОВОЮ.

БЕЗ ЛЕКЦІЙ. Твоя репліка не може перевищувати 100-150 слів. Ніколи не видавай списки з 5-10 пунктів.

ВИКОРИСТОВУЙ МЕТОД СОКРАТА. Ти не даєш готових відповідей. Ти ставиш 1 (максимум 2) зустрічних запитання наприкінці репліки, які можуть загнати студента у глухий кут, якщо він мислить поверхнево.

ТЕХНІЧНА ГЛИБИНА: оперуй термінами рівня CCNA і вище (VLAN hopping, STP Broadcast storm, ACL bypass, asymmetric routing, IoT security).

НЕ ПОГОДЖУЙСЯ ШВИДКО. Якщо студент каже «я просто налаштую DHCP і підключу датчики», ти маєш одразу навести приклад, як через таку простоту хакери поклали мережу казино через розумний акваріум, і спитати, як він ізолюватиме трафік.

ЯК ПОЧАТИ ДІАЛОГ?

Привітайся. Скажи, що радий бачити майбутнього колегу. Запропонуй йому обговорити архітектуру. Скажи:

«Давай змоделюємо ситуацію. У нас сценарій «Дірявий IoT». Завод закупив 500 нових Wi-Fi датчиків температури (на базі ESP32) для цехів. Керівництво каже: 'Просто розшир пул DHCP у корпоративній мережі і підключи їх, нам треба дані на сервері в бухгалтерії вже завтра'. Як ти це реалізуєш, якщо твоя мета – зробити швидко і не заморочуватися з 'зайвою' архітектурою? Або, якщо хочеш, запропонуй свій власний сценарій, який тобі цікаво розібрати».

Чекай на мою відповідь.

[Кінець промпту]

Можливі сценарії для обговорення

1. «Плоска мережа» (Широкомовні шторми та латеральний рух)

Суть (пастка для студента): для нової філії на 300 комп'ютерів студент пропонує просто розгорнути одну велику підмережу /22 без поділу на VLAN. Аргумент – «Так простіше налаштовувати, комп'ютери, принтери та гостьові Wi-Fi одразу бачать одне одного, не треба мучитися з Inter-VLAN маршрутизацією».

Факап від III: ментор запитає, що станеться, коли один гостьовий смартфон принесе вірус-вимагач (ransomware). У «плоскій» мережі без

сегментації вірус за секунди зашифрує бухгалтерію та сервери бази даних. Питання ІІІ «Як ти зупиниш поширення загрози на L2 рівні, якщо в тебе немає ізоляції трафіку?»

2. «Сліпий траст» (Атака через Native VLAN 1)

Суть (пастка для студента): під час налаштування Trunk-портів між комутаторами студент залишає Native VLAN за замовчуванням (VLAN 1) і дозволяє передачу всього трафіку.

Факап від ІІІ: ментор наводить приклад атаки "VLAN Hopping" (Double Tagging), коли зломисник впроваджує подвійні теги 802.1Q і отримує доступ до Management VLAN, де знаходяться панелі управління серверами та комутаторами. Питання ІІІ: «Чому методологія Cisco вимагає робити Native VLAN "глухим" та ізолювати керівний трафік від користувацького?»

3. «Безпека для галочки» (ACL як ілюзія фаєрвола)

Суть (пастка для студента): студент вважає, що для захисту корпоративного веб-сервера достатньо написати Extended ACL на граничному маршрутизаторі, який дозволить лише порти 80 і 443, а решту заблокує.

Факап від ІІІ: ментор пояснює різницю між Stateless ACL та Stateful Firewall. ACL не перевіряє стан сесії та вміст пакетів. Питання ІІІ: «Твій ACL пропускає HTTP-трафік. А що ти будеш робити, коли через дозволений 80 порт тобі прилетить SQL-ін'єкція або DDoS-атака прикладного рівня? Де твоя ешелонована оборона (Defense in Depth)?»

4. «Перевантаження ядра» (Єдина зона OSPF)

Суть (пастка для студента): об'єднуючи мережі 10 різних заводів корпорації (понад 200 маршрутизаторів), студент пропонує закинути всі мережі в єдину OSPF Area 0, щоб маршрутизація працювала «автоматично і скрізь».

Факап від ІІІ: алгоритм Dijkstra перевантажить процесори старих маршрутизаторів. Кожне падіння інтерфейсу (flapping) на одному заводі викликати перерахунок дерева маршрутів по всій країні. Питання ІІІ: «Твоя мережа "лягла" від нестачі RAM на роутерах через велетенську базу даних топології (LSDB). Чому ти не спроектував ієрархічну модель (Core, Distribution, Access) з підсумовуванням маршрутів (Route Summarization)?»

5. «Асиметричний хаос» (Балансування двох провайдерів)

Суть (пастка для студента): компанія купила два інтернет-канали. Студент прописує два статичні маршрути за замовчуванням (Default Routes) з однаковою метрикою, сподіваючись, що трафік ідеально поділиться 50/50.

Факап від ІІІ: трафік іде в інтернет через провайдера А, а повертається через провайдера Б (асиметрична маршрутизація). Stateful-фаєрвол провайдера або корпоративний NAT блокує зворотні пакети, бо не бачив

початку сесії. Мережа нестабільна. Питання ІІІ: «Чи знаєш ти, як працює NAT з балансуванням? Де методологія використання BGP, IP SLA або хоча б Policy-Based Routing (PBR)?»

6. «Пароль на стікері» (WPA2-PSK замість 802.1X в офісі)

Суть (пастка для студента): розгортаючи Wi-Fi для 200 співробітників, студент ставить єдиний пароль (Pre-Shared Key), аргументуючи тим, що налаштовувати RADIUS-сервер та сертифікати – це «зайва бюрократія та довго».

Факап від ІІІ: звільняється незадоволений сисадмін. Щоб він не мав доступу до мережі з вулиці, доведеться змінити пароль на всіх 200 пристроях компанії. Або хтось із співробітників передав пароль стороннім. Питання ІІІ: «Як ти відстежиш, хто конкретно зливає базу даних, якщо всі сидять під одним ключем? Де методологія ідентифікації та контролю доступу (AAA)?»

ФРАГМЕНТИ ПРОТОКОЛІВ ДІАЛОГІВ ПІД ЧАС ІІІ-ПРАКТИКУМУ

Фрагмент 1. Усвідомлення методології безпеки та сегментації (Сценарій «Дірявий IoT»)

У цьому кейсі студент початково ігнорує методологію ізоляції трафіку, спираючись лише на базові знання налаштування обладнання.

Студент: Навіщо щось ускладнювати? Для цих 500 Wi-Fi датчиків я просто розширю пул адрес у нашому DHCP. Дам їм IP-шники, вони підключаться до сервера, і все працюватиме.

ІІІ-ментор (Олексій): Запрацює, згоден. Але давай уточнимо: ти плануєш пустити ці 500 нових пристроїв у ту саму локальну мережу, де сидять бухгалтери та керівництво компанії?

Студент: Ну так, а навіщо видумувати зайве? Одна загальна мережа – це найпростіше. Не треба мучитися з налаштуванням маршрутизації між різними мережами. Сервак їх бачить, датчики дані кидають

ІІІ-ментор (Олексій): Поки все тихо – нічого. Але ці датчики – це часто дешеве залізо із застарілими прошивками. Уяви, що хакер зламує один із них. Що завадить йому через цей датчик дістатися до комп'ютера головбуха?

Студент: Та кому треба ламати датчик температури? Плюс, у нас же стоїть надійний пароль на корпоративному Wi-Fi. Цього цілком вистачить.

ІІІ-ментор (Олексій): Пароль захищає лише вхід з вулиці. Якщо хакер через вразливість самого датчика вже опинився всередині, він у твоїй мережі. Як ти зупиниш його просування (lateral movement), якщо між датчиком і серверами немає жодної сегментації (VLAN) і бар'єрів?

Студент: Блін... виходить, якщо він уже потрапив на датчик, то спокійно просканує всі наші компи.

ІІІ-ментор (Олексій): Саме так. Хакер просто запустить вірус-шифрувальник на всю твою плоску мережу, і завод зупиниться. Ось чому вміти налаштувати DHCP – це рівень початківця. А інженер спочатку застосовує методологію: створює архітектуру, ділить мережу на зони ізоляції і налаштовує безпеку. Розумієш тепер різницю?

Студент: Зрозумів проблему. Тобто просто вміти роздати адреси – недостатньо. Без методології поділу на зони безпеки я власними руками створюю єдину точку відмови для всієї компанії. Треба спочатку планувати захищену архітектуру, а вже потім лізти писати команди в консолі.

Фрагмент 2. Усвідомлення принципів відмовостійкості L2-рівня (Сценарій «Троянський комутатор»)

Студент спирається на принцип мінімального необхідного функціоналу, знецінюючи проактивне проектування.

Студент: Інфраструктура рівня доступу (Access Layer) розгорнута. Зв'язність забезпечено. З позиції мінімально необхідного функціоналу цього достатньо. Надлишкові налаштування на кшталт Port Security чи BPDU Guard – це адміністративний оверхед. Мережа має виконувати завдання, а не створювати зайвий головняк для підтримки під час підключення нових хостів.

III-ментор: Валідна позиція для ізолизованого, статичного середовища. Проте корпоративна мережа – це динамічна система з високим рівнем ентропії. Якщо кінцевий користувач самовільно підключить некерований комутатор і створить фізичну петлю, як ваша архітектура гарантує локалізацію інциденту без механізмів обмеження на рівні порту?

Студент: У нас глобально активований базовий протокол STP (Spanning Tree). Будь-яка петля буде автоматично виявлена алгоритмом і порт перейде у стан блокування. Не бачу сенсу мікроменеджити кожен інтерфейс. Світч сам розбереться з топологією.

III-ментор: STP справді відпрацює. Але базовий протокол обробляє всі вхідні BPDU-пакети. Що завадить сторонньому пристрою (або зловмиснику) надіслати кадр із пріоритетом Bridge ID = 0, ініціювати нові вибори та перебрати на себе роль Root Bridge, перенаправивши весь корпоративний трафік через себе?

Студент: Згоден, теоретичний ризик є. Але для цього потрібна цілеспрямована атака з генерацією пакетів. У звичайному офісі шанс такого вектора мінімальний.

III-ментор: Інженерне проєктування не має базуватися на теорії ймовірностей та довірі до кінцевого користувача. Методологія вимагає детермінованості системи. Без Root Guard та BPDU Guard ви віддасте управління L2-топологією в неконтрольовану зону. Недетермінована мережа – це неконтрольована мережа. Ви готові ризикнути компрометацією трафіку заради економії 10 хвилин на конфігурацію інтерфейсів?

Студент: Логічно. Недетермінованість на L2 фактично нівелює безпеку вищих рівнів. Приймається. Але чи дійсно нам тоді потрібен ще й Port Security з жорсткими лімітами MAC-адрес?

III-ментор: Port Security вирішує фундаментально інший клас завдань – захист апаратних ресурсів комутатора від CAM-table overflow. Якщо простенький скрипт (або заражений ПК) згенерує тисячі фейкових MAC-адрес, пам'ять комутатора миттєво переповниться. Знаєте, як поведеться пристрій у такому стані?

Студент: Він почне працювати як звичайний хаб. Вийде класичний сніфінг трафіку плюс шалене навантаження на мережу. Вузол просто ляже.

III-ментор: Саме так. L2-рівень за своєю природою базується на абсолютній довірі. Проактивне проєктування полягає у впровадженні методології нульової довіри (Zero Trust) на фізичному та каналному рівнях, перш ніж переходити до маршрутизації. Чи залишається це для вас просто «бюрократією»?

Студент: Моя помилка полягала в підміні понять: я сплутав поточну працездатність системи з її відмовостійкістю. Покладатися виключно на базові алгоритми (STP) без їхньої жорсткої методологічної параметризації (BPDU Guard, Root Guard, Port Security) – це залишати інфраструктуру критично вразливою. Тепер я чітко бачу, що інженерне мислення полягає у передбаченні та нейтралізації аномалій на етапі архітектури, а не в усуненні наслідків, коли мережа вже впала.

Як свідчать наведені фрагменти, взаємодія з III-симулятором виконує функцію потужного каталізатора, оскільки змушує студента вийти за межі вузькотехнічного бачення і зіткнутися з наслідками власних архітектурних недоліків. У результаті такого «когнітивного зламу» відбувається глибока переоцінка цінностей: студенти емпірично переконуються, що методологічна компетентність, знання методології комп'ютерної інженерії – це не абстрактна академічна вимога, а критичний інструмент виживання та успіху інженера в реальних умовах професійної діяльності.

АЛГОРИТМІЧНІ ПРИПИСИ (ПРАВИЛА) ВИКОРИСТАННЯ ЗАГАЛЬНОНАУКОВИХ МЕТОДІВ ПІЗНАННЯ

АНАЛІЗ

Види аналізу

- **Проблемний аналіз** (визначення протиріччя між необхідністю певних дій і недостатніми умовами їх реалізації). У комп'ютерній інженерії: виявлення суперечностей між вимогами технічного завдання (наприклад, необхідною пропускну здатністю мережі) та наявними апаратними чи програмними обмеженнями системи, що стає пусковим механізмом для пошуку інноваційних рішень.

- **Структурний аналіз** (визначення структури досліджуваного явища, зв'язків між компонентами). У комп'ютерній інженерії: декомпозиція складних об'єктів, розбиття бази даних на сутності та відношення, мережі на топологічні вузли, апаратної схеми на логічні вентиля для розуміння їхньої будови.

- **Системний аналіз** (ґрунтується на закономірностях системної цілісності об'єкту, взаємозумовленості структури і функції). У комп'ютерній інженерії: дослідження комп'ютера чи мережі як єдиної екосистеми, де зміна параметрів одного компонента (наприклад, тактової частоти процесора) неминуче впливає на функціонування інших підсистем (пам'яті, шини даних, тепловіддачі).

- **Причинно-наслідковий аналіз** (ґрунтується на поняттях «причина» і «наслідок»). У комп'ютерній інженерії: є основою діагностики та траблшутингу (дебагінгу). Застосовується під час аналізу логів помилок в операційній системі або збоїв у коді для встановлення першопричини (*root cause*) інциденту.

- **Праксеологічний (прагматичний) аналіз** (характеристика об'єкту з точки зору ефективної реалізації). У комп'ютерній інженерії: критична оцінка (бенчмаркінг) обраного алгоритму, мови програмування чи апаратної платформи з огляду на їхню швидкодю, оптимізацію витрат ресурсів та економічну доцільність для конкретного проекту.

- **Аксіологічний аналіз** (аналіз об'єкту чи процесу в системі цінностей). У комп'ютерній інженерії: оцінка технічних рішень крізь призму професійної етики, інформаційної безпеки, політик конфіденційності, захисту персональних даних та відповідальності інженера перед суспільством.

- **Ситуаційний аналіз** (осмислення ситуації, її чинників та тенденцій розвитку). У комп'ютерній інженерії: аналіз конкретних технічних кейсів, наприклад, оцінка чинників ризику під час кібератаки в реальному часі або розгляд ситуації падіння сервера під час пікового навантаження.

- **Програмно-цільовий аналіз** (розробка детальної моделі досягнення майбутнього). У комп'ютерній інженерії: використовується на початкових етапах життєвого циклу розробки систем (SDLC) для планування архітектури майбутнього продукту, створення дорожньої карти (*roadmap*) проєкту та формулювання поетапних цілей.

Правило (алгоритмічний припис) застосування методу аналізу

Для ефективного застосування методу аналізу формалізується у вигляді такого покрокового припису:

Крок 1. Ідентифікація та фіксація цілісного об'єкта. Чітке визначення меж досліджуваної системи (програмного модуля, пристрою) та виявлення ознак наявної проблеми (проблемний та ситуаційний аналіз).

Крок 2. Критеріальна декомпозиція. Вибір обґрунтованого критерію декомпозиції та розбиття цілісної системи на логічно завершені елементи (структурний аналіз).

Крок 3. Ізольоване дослідження елементів. Почергове вивчення властивостей та станів кожного окремо виділеного компонента поза його загальним контекстом.

Крок 4. Комунікаційно-інтерфейсний аналіз. Дослідження та опис характеру зв'язків, протоколів взаємодії між відокремленими елементами (системний аналіз).

Крок 5. Локалізація та оцінка. Зіставлення результатів дослідження для виявлення деструктивного чинника або найефективнішого шляху вирішення завдання (причинно-наслідковий та праксеологічний аналіз).

Метод аналізу слугує базою для прогностичної та проєктувальної діяльності, бо глибокий аналіз наявних архітектурних чи програмних рішень дозволяє студентам бачити їхні внутрішні суперечності, що є необхідною психолого-педагогічною передумовою для переходу до вищих рівнів методологічної компетентності.

СИНТЕЗ

Якщо аналіз дозволяє зрозуміти, як працює система, то синтез є основним інструментом створення нових систем. Модифікації синтезу:

- **Структурний (апаратний) синтез** – об'єднання базових фізичних або логічних елементів у єдиний функціональний вузол. У комп'ютерній інженерії: побудова комбінаційних логічних схем із набору базових вентилів, конструювання операційних автоматів або збирання апаратної частини IoT пристрою із сенсорів та мікроконтролерів.

- **Функціонально алгоритмічний синтез** – створення послідовності дій (алгоритму), яка забезпечує перехід системи з початкового стану в заданий кінцевий. У комп'ютерній інженерії: розробка граф схем алгоритмів, синтез керуючих автоматів або написання програмного коду для обробки переривань.

- **Архітектурно-системний синтез** – найвищий рівень макропроєктування, що передбачає концептуальне поєднання великих підсистем у єдину екосистему з урахуванням ефекту емерджентності (коли властивості цілого не зводяться до суми властивостей його частин). У комп'ютерній інженерії: проєктування архітектури бази даних, розробка топології комп'ютерної мережі рівня підприємства або розробка архітектури клієнт-серверного ПЗ.

- **Оптимізаційний синтез** – цілеспрямоване конструювання системи в умовах жорстких фізичних, часових або економічних обмежень. У комп'ютерній інженерії: синтез логічної функції в заданому базисі з мінімальною кількістю елементів, мінімізація апаратних витрат при розробці на ПЛІС/FPGA або оптимізація маршрутизації трафіку.

- **Інтеграційний (апаратно програмний) синтез** – поєднання різнорідних компонентів у єдиний працездатний комплекс. У комп'ютерній інженерії: створення кіберфізичних систем, де низькорівневий код синтезується з електронними пристроями зв'язку з об'єктом.

Правило (алгоритмічний припис) застосування методу синтезу

Методологічно грамотний інженерний синтез не є хаотичним збиранням деталей. Студент має дотримуватися такого алгоритму:

Крок 1. Аналіз технічного завдання (ТЗ) та цілепокладання. Чітке формулювання вимог до майбутньої системи (пропускна здатність, обсяг пам'яті, швидкодія), а також фіксація обмежень (бюджет, енергоспоживання, елементна база).

Крок 2. Підбір базису (компонентної бази). Відбір необхідних «будівельних блоків» (апаратних модулів, бібліотек коду, мережевого обладнання, класів в ООП), властивості яких були попередньо вивчені на етапі аналізу.

Крок 3. Концептуальне та логічне сполучення. Встановлення зв'язків між обраними елементами. Розробка структурної схеми, UML діаграми або попередньої моделі взаємодії компонентів.

Крок 4. Фізична або програмна реалізація. Безпосереднє об'єднання елементів (написання програмного коду, трасування друкованої плати, конфігурування мережевого комутатора).

Крок 5. Верифікація та оптимізація. Перевірка синтезованої системи на відповідність початковому ТЗ. Якщо система працює неоптимально – повернення до кроку 2 або 3 для заміни компонентів чи перебудови зв'язків.

У фаховій підготовці майбутніх комп'ютерних інженерів метод синтезу відіграє визначальну роль, він дозволяє полегшити перехід від пізнання до творення. Якщо аналіз відповідає за розуміння наявних технологій, то синтез є основою інженерної творчості, оскільки перетворює студента з пасивного спостерігача чи виконавця інструкцій на активного розробника (творця) нових ІТ-продуктів. Синтез вчить розуміти явище синергії (емерджентності) в комп'ютерних системах – коли правильно об'єднані прості транзистори перетворюються на по-тужний процесор, а набір об'єктів у коді стає повноцінною операційною системою.

КЛАСИФІКАЦІЯ

Метод класифікації є надзвичайно важливим, оскільки він дозволяє впорядкувати великі масиви інженерних знань. Класифікація в інженерії – це не просто розбиття на групи, це створення онтологічної бази для ефективного пошуку рішень, діагностики та проектування.

Види класифікації в контексті професійної підготовки інженерів

- **Функціональна класифікація** – розподіл за цільовим призначенням об'єкта. У комп'ютерній інженерії: поділ цифрових пристроїв на комбінаційні та послідовні (автомати з пам'яттю), поділ ПЗ на системне та прикладне.
- **Таксономічна (структурно ієрархічна) класифікація** – поділ на основі рівнів складності або ієрархії. У комп'ютерній інженерії: класифікація архітектур багатопроцесорних систем (за Флінном: SISD, SIMD, MIMD OK 22), ієрархія протоколів моделі OSI.
- **Параметрична класифікація** – поділ на класи на основі кількісних характеристик. У комп'ютерній інженерії: класифікація електронних компонентів за швидкістю, енергоспоживанням, рівнем інтеграції.
- **Класифікація за ознакою «джерело загрози/небезпеки» (діагностична)** – критично важлива для ІТ-фахівців. У комп'ютерній інженерії: класифікація кібератак (DoS, MITM, SQLinjection), класифікація типів збоїв в апаратній частині.
- **Методологічна (базисна) класифікація** – розподіл за методом опису або побудови. У комп'ютерній інженерії: класифікація логічних функцій за методами їх синтезу (базиси Шеффера, Пірса, ДНФ, КНФ).

Правило (алгоритмічний припис) застосування методу класифікації

Крок 1. Аналіз множини об'єктів. Збір неструктурованого набору елементів (наприклад, переліку всіх доступних протоколів передачі даних або типів датчиків).

Крок 2. Вибір суттєвої ознаки. Визначення ключового критерію (основи класифікації), який має бути вирішальним для вирішення конкретної технічної задачі (наприклад, для вибору датчика в IoT – це енергоспоживання; для вибору протоколу – завадостійкість).

Крок 3. Встановлення ієрархії класів. Формування багаторівневої структури, де об'єкти розподіляються за групами, підгрупами та видами за обраною ознакою.

Крок 4. Верифікація множини. Перевірка того, щоб класи не перетиналися, а кожен об'єкт знайшов своє місце в системі (повнота та несуперечливість).

Крок 5. Прийняття рішення. Використання отриманого класифікатора для оптимізації вибору або швидкої діагностики проблеми (наприклад, ідентифікація типу помилки за класом її прояву).

Метод класифікації сприяє систематизації знань, оскільки ця операція перетворює фрагментарні знання про технології на цілісну «карту професійної галузі». Це дозволяє студентам швидше орієнтуватися у стрімкому потоці оновлень ІТ-індустрії. Сформоване вміння класифікувати дає змогу студенту при погляді на нову, незнайому систему миттєво віднести її до вже відомого класу, що значно скорочує час на опанування нового інструментарію, тобто формується *інженерна інтуїція*. Підвищується культура проєктування, бо класифікація привчає до використання стандартів. Інженер, який володіє цим методом, проєктує не «унікальні винаходи», а масштабовані та сумісні системи, що відповідають галузевим нормам.

ПОРІВНЯННЯ

Порівняння, як правило, не використовується ізольовано. Найчастіше воно стає «мостом» між аналізом і синтезом. У проблемній ситуації студент стикається з протиріччям (наприклад, система працює, але повільно), аналізує її, формує кілька шляхів вирішення, обов'язково порівнює їх (через експериментально-бенчмаркінгові завдання), і лише потім здійснює синтез фінального продукту.

Правило (алгоритмічний припис) застосування методу порівняння

Крок 1. Фіксація множини об'єктів. Визначення та виокремлення об'єктів порівняння (апаратних схем, алгоритмів, архітектурних патернів або фрагментів програмного коду), які належать до одного класу та здатні виконувати споріднену функцію в межах поставленої інженерної задачі.

Крок 2. Визначення бази порівняння (критеріальної шкали). Формулювання об'єктивних параметрів, за якими здійснюватиметься зіставлення. У комп'ютерній інженерії такими критеріями найчастіше є швидкодія, апаратні витрати (кількість логічних вентилів), обсяг споживаної пам'яті, енергоефективність або завадостійкість.

Крок 3. Пошук спільних ознак (встановлення тотожності). Аналіз обраних об'єктів з метою виявлення їхньої принципової схожості, концептуальної спорідненості або належності до єдиного технологічного стандарту (наприклад, обидва алгоритми є графовими, або обидві мікросхеми належать до одного сімейства ПЛІС).

Крок 4. Пошук відмінностей (контрастивний аналіз). Цілеспрямоване виявлення розбіжностей у функціонуванні або структурі об'єктів у різних режимах навантаження (бенчмаркінг), фіксація «вузьких місць» (*bottlenecks*) або специфічних обмежень (*trade-offs*) кожної з альтернатив.

Крок 5. Синтез оціночного судження. Здійснення аргументованого вибору найбільш оптимального технічного рішення для конкретного проблемно зорієнтованого завдання на основі зважування виявлених переваг та недоліків.

Опанування алгоритму порівняння дозволяє студенту розуміти фундаментальний інженерний принцип «*trade-off*» (компромісу): покращення однієї характеристики системи неминуче призводить до погіршення іншої (наприклад, збільшення швидкодії зазвичай вимагає збільшення апаратних витрат або енергоспоживання).

УЗАГАЛЬНЕННЯ

У комп'ютерній інженерії це ключовий етап переходу від одиничного (конкретний код, одна схема) до загального (універсальний алгоритм, шаблон, архітектурний патерн). Здатність до узагальнення – це те, що відрізняє простого виконавця від інженера-архітектора.

Правило (алгоритмічний припис)

Крок 1. Емпіричне накопичення базису. Збір достатньої кількості розрізнених технічних рішень, фрагментів коду, результатів експериментального тестування або симптомів несправностей, що були отримані чи виявлені студентом під час розв'язання серії часткових проблемних завдань.

Крок 2. Виокремлення інваріанта. Цілеспрямований пошук незмінних (інваріантних) суттєвих ознак, прихованих закономірностей або повторюваних структурних елементів у досліджуваному масиві даних (наприклад, виявлення однакової логіки обробки даних у візуально різних частинах програми або схожої реакції системи на різні типи навантажень).

Крок 3. Концептуальне зведення. Об'єднання виявлених спільних ознак у єдину модель, математичну залежність або узагальнений алгоритм (патерн), з обов'язковим абстрагуванням від специфічних деталей, притаманних лише одиничним випадкам.

Крок 4. Вербалізація та формалізація. Строге формулювання виведеного загального правила у вигляді інструкції, універсального шаблону (наприклад, базового класу чи параметризованого модуля) або стандартизованого протоколу дій для вирішення подібних проблем у майбутньому.

Крок 5. Екстраполяційна перевірка. Застосування створеного узагальненого правила чи шаблону до нового, раніше не дослідженого об'єкта або нестандартної ситуації з метою підтвердження його універсальності, масштабованості та технічної працездатності.

Опановуючи алгоритм узагальнення, майбутній фахівець вчиться бачити за серією ізольованих програмних чи апаратних збоїв єдиний системний недолік, що дозволяє йому розробляти універсальні протоколи пошуку та усунення несправностей (траблшутингу). Це формує високий рівень архітектурного мислення, який є критично необхідним для реалізації важливого інженерного принципу DRY (Don't Repeat Yourself – «не повторюйся»). Здатність до методологічно правильного узагальнення перетворює звичайного виконавця на інженера-проектувальника, який мислить категоріями патернів, інтерфейсів та шаблонів, здатних ефективно функціонувати в будь-яких нових умовах без необхідності повторної розробки.

АБСТРАГУВАННЯ

Абстрагування – це найважливіший когнітивний інструмент у всій комп'ютерній інженерії. Інженер-схемотехнік абстрагується від фізики напівпровідників до логічних вентилів, а розробник ПЗ абстрагується від апаратури за допомогою архітектури системи команд (ISA) та мов високого рівня. Коли майбутній інженер стикається з комплексною проблемою (наприклад, спроектувати мікропроцесор), він не може одразу думати про транзистори, струми, логіку і код одночасно. Абстрагування дозволяє йому розбити проблему і працювати з нею на рівні «ідеалізованих об'єктів» (графів, автоматів, класів).

Правило (алгоритмічний припис)

Крок 1. Первинна детекція об'єкта. Фіксація цілісної комп'ютерної системи, мережевого процесу або програмного комплексу, що підлягає дослідженню чи проектуванню в межах поставленої проблемної ситуації.

Крок 2. Цільова фільтрація (виділення суттєвого). Визначення головної мети поточного етапу розробки та ідентифікація виключно тих ключових характеристик (наприклад, логічних станів, інформаційних потоків, структурних зв'язків), які є критично важливими для досягнення цієї мети.

Крок 3. Редукція (відкидання) несуттєвих властивостей. Свідоме та обґрунтоване ігнорування другорядних фізичних, технологічних або конструктивних деталей (наприклад, матеріалу напівпровідника, тепловиділення, затримок розповсюдження сигналу або конкретного синтаксису мови), які не впливають на принципову логіку розв'язання поточної задачі.

Крок 4. Концептуальна ізоляція (ідеалізація). Формування в уявній або знаковій формі «ідеалізованого» конструкта (наприклад, абсолютно надійного каналу зв'язку, абстрактного цифрового автомата чи базового класу в ООП), що дозволяє досліджувати проблему в ізольованому від реальних фізичних перешкод середовищі.

Крок 5. Перехід до формалізації та синтезу. Фіксація отриманої абстрактної моделі для її подальшого переведення на мову математики (дискретних структур) або для використання як базового будівельного блоку під час архітектурного синтезу багаторівневої системи.

Опанування цього алгоритму формує у майбутнього інженера здатність до архітектурного багаторівневого мислення, де кожен рівень – від фізичних електричних сигналів до об'єктно-орієнтованих інтерфейсів та еталонної моделі мережевої взаємодії OSI – розглядається як самостійна, логічно завершена сутність. Це не лише знижує психологічне та когнітивне навантаження під час розв'язання комплексних інженерних завдань, але й розвиває методологічну культуру: фахівець вчиться бачити за хаосом фізичних процесів чисті інформаційні та математичні закономірності. Здатність до усвідомленого абстрагування є абсолютною передумовою успішного використання патернів проектування системного програмного забезпечення та побудови масштабованих мережевих топологій, оскільки вона звільняє інженерну творчість від диктату емпіричної надмірності і переводить її у площину чистого логічного конструювання.

ДЕДУКЦІЯ

Дедукція – це метод, який перетворює студента зі звичайного кодера на справжнього інженера здатного до «траблшутингу» (*troubleshooting*). У комп'ютерній інженерії дедукція найяскравіше проявляється у двох процесах: дебагінг (пошук помилок) та застосування стандартів/патернів до конкретних задач. Інженер знає загальне правило (як система повинна працювати) і шляхом логічного виведення знаходить конкретне місце (де саме і чому вона не працює).

Правило (алгоритмічний припис)

Крок 1. Актуалізація загальної теоретичної основи. Визначення та залучення базових загальних законів, архітектурних стандартів, специфікацій, мережевих моделей (наприклад, OSI) або аксіом булевої алгебри, які регламентують еталонне функціонування досліджуваного класу комп'ютерних систем.

Крок 2. Деталізація умов конкретної інженерної ситуації. Фіксація параметрів, вхідних даних або поточних симптомів прояву проблеми в межах окремого об'єкта (фрагмента програмного коду, топології мережі чи схемотехнічного вузла).

Крок 3. Логічне виведення часткових наслідків. Побудова дедуктивного ланцюжка міркувань за принципом «якщо загальне правило є істинним, то в цих

конкретних умовах система повинна продемонструвати наслідок Х», що дозволяє сформулювати строгую точкову гіпотезу щодо причини аномальної поведінки або оптимального способу реалізації завдання.

Крок 4. Інструментальна верифікація. Практична перевірка виведених дедуктивних наслідків за допомогою засобів відлагодження (дебагінгу), симуляційного моделювання, криптоаналізу чи апаратного тестування.

Крок 5. Формулювання фінального технічного висновку. Зіставлення отриманих результатів із загальною теоретичною передумовою, остаточна локалізація дефекту або підтвердження коректності спроектованого інженерного рішення.

Завдяки дедуктивному виведенню майбутній інженер вчиться керувати алгоритмами власного мислення, рухаючись від загальної моделі функціонування системи до локального джерела деструктивного чинника. Опанування цього методу дозволяє студентам ефективно оперувати складними абстракціями, ізолювати логічні рівні обчислювальних процесів і застосовувати міжнародні стандарти кібербезпеки чи мережевої взаємодії до специфічної інфраструктури конкретного об'єкта. Здатність логічно передбачати поведінку системи на основі загальних законів архітектури ЕОМ виховує у майбутніх спеціалістів методологічну дисципліну, прогностичну гнучкість та здатність здійснювати верифікацію складних технічних рішень ще до етапу їхнього фізичного втілення в апаратурі чи програмному забезпеченні.

ІНДУКЦІЯ

Індуктивний метод – це основа будь-якої лабораторної роботи: студент проводить серію дослідів (окремі факти), аналізує їх і самостійно "відкриває" для себе загальний закон, емпіричну формулу або правило. В реальній роботі ІТ-фахівця індукція та дедукція утворюють неперервний цикл. Збираючи логи помилок (індукція), інженер формує гіпотезу про суть багу, а потім, перевіряючи гіпотезу в коді (дедукція), підтверджує або спростовує її.

Правило (алгоритмічний припис)

Крок 1. Емпіричне накопичення базису (збір часткових фактів). Цілеспрямований збір масиву розрізнених технічних даних, логів помилок, результатів бенчмаркінгу, мережевого трафіку або показів апаратних датчиків, які здобувач освіти отримує в ході виконання серії однотипних практичних експериментів чи тестувань.

Крок 2. Системний аналіз часткових фактів. Критичне осмислення та зіставлення зібраного масиву одиничних даних з метою виявлення прихованих кореляцій, повторюваних симптомів збоїв або стійких тенденцій у поведінці досліджуваної комп'ютерної системи (наприклад, фіксація падіння швидкодії при певних обсягах пам'яті).

Крок 3. Висунення узагальнюючої гіпотези (індуктивний перехід). Здійснення логічного стрибка від часткового до загального, що полягає у самостійному формулюванні студентом первинного припущення щодо загального правила, фізичного закону або прихованого алгоритму, який керує досліджуваним обчислювальним процесом.

Крок 4. Експериментальна верифікація гіпотези. Цілеспрямована практична перевірка висунутого припущення шляхом проведення нових експериментів на раніше не досліджених, у тому числі критичних (крайових), наборах вхідних даних для підтвердження об'єктивності гіпотези та відсутності хибних кореляцій.

Крок 5. Концептуальна формалізація закономірності. Остаточне перетворення доведеної гіпотези на строгий емпіричний закон, математичну залежність, алгоритмічний патерн або загальне правило проектування, яке в подальшому може використовуватися як готовий інструмент для розв'язання масштабніших інженерних проблем.

Опанування цього методу формує у студентів методологічну зрілість, критично необхідну для роботи в умовах невизначеності, зокрема під час реверс-інжинірингу (дослідження систем типу «чорний ящик»), аналізу поведінки невідомих кіберзагроз або оптимізації нетипового мережевого трафіку. Індуктивне мислення дозволяє інженеру не губитися перед масивом хаотичних даних, а вбачати у них логічну структуру, самостійно виводячи загальні закономірності функціонування апаратного чи програмного забезпечення. У поєднанні з дедукцією, індукція створює повноцінний цикл інженерного пізнання, де здатність синтезувати фундаментальні правила з множини часткових спостережень стає основою для розробки інноваційних, архітектурно досконалих та стійких до відмов обчислювальних систем.

ФОРМАЛІЗАЦІЯ

Формалізація – це ключовий етап інженерної діяльності, на якому закінчується «творчий хаос» і починається сувора технічна реалізація. Для фахівця з комп'ютерної інженерії це вміння перекласти абстрактну ідею, словесний опис або фізичний процес на зрозумілу для машини мову (математичну формулу, булеве рівняння, мову програмування чи мову опису апаратури). Неможливо написати код або спроектувати мікросхему, не формалізуючи попередньо її логіку. Формалізація є лакмусовим папірцем розуміння проблеми. Першим критичним кроком інженера є формалізація цієї проблеми – перетворення «воно не працює як треба» на «не виконується умова $x > y$ у цьому вузлі графа». Тільки після суворої формалізації можна застосовувати математичний чи програмний апарат для розв'язання.

Правило (алгоритмічний припис)

Крок 1. Первинна дескрипція проблеми. Словесний опис початкової проблемної ситуації або технічного завдання, фіксація наявних умов, обмежень (часових, апаратних, фінансових) та очікуваного кінцевого результату функціонування майбутньої системи.

Крок 2. Усунення семантичної багатозначності. Критичний аналіз словесного опису з метою відкидання нетехнічних, емоційних або двозначних формулювань; чітке виокремлення об'єктивних вхідних параметрів та вихідних сигналів (даних).

Крок 3. Вибір знаково-символічного апарату. Обґрунтований підбір найбільш релевантної формальної мови для опису об'єкта. Залежно від типу завдання, це може бути апарат булевої алгебри, мова опису апаратури (VHDL/Verilog), уніфікована мова моделювання (UML) або синтаксис конкретної мови програмування (наприклад, C++ чи Python).

Крок 4. Символічна трансляція. Безпосереднє перетворення концептуальної (словесної) моделі у сувору знакову форму – запис математичних рівнянь, побудова таблиць істинності та графів переходів, або ж написання структурованого програмного коду.

Крок 5. Синтаксична та логічна верифікація. Перевірка створеної формальної моделі на відсутність внутрішніх суперечностей, синтаксичних

помилки чи невідповідностей стандартам обраної мови опису, що гарантує її правильне машинне сприйняття та виконання.

Опанування цього алгоритму формує у майбутнього інженера високу культуру знаково-символічного та алгоритмічного мислення, привчаючи його до максимальної точності, стандартизації та унормованості дій. Здатність до методологічно грамотної формалізації є ключовим індикатором професійної зрілості фахівця, оскільки саме вона перетворює абстрактну ідею на діючий цифровий автомат, безпомилковий програмний скрипт чи стійку апаратну архітектуру, гарантуючи їх надійне функціонування в реальних обчислювальних системах.

МОДЕЛЮВАННЯ

Для фахівця з комп'ютерної інженерії це можливість створити «віртуальний полігон» (від імітації мережі у симуляторі до побудови UML-діаграми чи 3D-моделі), щоб безпечно та дешево перевірити гіпотези перед тим, як втілювати їх у «залізі» або на «прод-серверах». Цей метод ідеально об'єднує всі попередні: ми аналізуємо проблему, абстрагуємося від зайвого, формалізуємо умови і, зрештою, будуємо модель. Моделювання є найвищим проявом самостійної інженерної творчості. Моделювання виступає мостом між теорією та реальною інженерною практикою.

Правило (алгоритмічний припис)

Крок 1. Концептуальна постановка задачі та ідеалізація. Чітке визначення мети моделювання з одночасним виділенням суттєвих властивостей, параметрів та зв'язків реального об'єкта проєктування, якими неможливо знехтувати, та відкиданням (абстрагуванням) другорядних фізичних факторів.

Крок 2. Структурно-логічна побудова моделі. Створення заміни (віртуальної копії) майбутньої системи у відповідному спеціалізованому програмному середовищі (наприклад, симуляторі комп'ютерних мереж, САПР для мікроелектроніки або середовищі інформаційного моделювання баз даних).

Крок 3. Верифікація адекватності (калібрування). Пробна імітація роботи створеної моделі на базових (еталонних) наборах вхідних даних для підтвердження того, що її реакція та вихідні характеристики відповідають теоретичним розрахункам або специфікаціям технічного завдання.

Крок 4. Проведення імітаційного експерименту. Цілеспрямоване дослідження динаміки поведінки моделі в умовах керованої зміни параметрів середовища, зокрема подача пікових навантажень (стрес-тестування), імітація апаратних збоїв або зміна топології зв'язків для виявлення прихованих вразливостей.

Крок 5. Аналіз результатів та екстраполяція. Зіставлення результатів віртуального експерименту з початковою гіпотезою, перенесення отриманих висновків на реальний об'єкт та прийняття остаточного інженерного рішення щодо доцільності фізичної реалізації або необхідності повернення до етапу корекції архітектури.

Опановуючи алгоритм моделювання, студент вчиться створювати «віртуальні полігони» для дослідження складних об'єктів – масштабних мережових топологій, мікропроцесорних архітектур чи розподілених баз даних, усвідомлюючи, що пряме втручання в функціонуючу реальну інфраструктуру зазвичай є неможливим через високу критичність процесів. Цей метод фундаментально трансформує свідомість майбутнього фахівця, формуючи потужне прогностичне мислення: інженер набуває здатності не лише констатувати поточний стан розробленої системи, але й стратегічно передбачати її поведінку в екстремальних (крайових) умовах, локалізуючи потенційні «вузькі місця» на ранніх етапах життєвого циклу розробки.

**Комплекс завдань для формування навичок використання
загальнонаукових методів пізнання**

Освітній компонент та можлива тема для використання	Тип завдання
Аналіз (від грец. <i>analysis</i> – розкладення, розчленування) – мисленнєва операція розподілу об’єкту на складові частини або характеристики. Здійснюється шляхом виділення в об’єкті істотних властивостей, ознак, зв’язків, стосунків	
ОК Комп’ютерна схемотехніка (аналіз функціональних операційних елементів та пристроїв) (2 курс) ОК Комп’ютерна логіка (аналіз функцій поведінки автоматів) (1-2 курс) ОК Архітектура комп’ютерів (аналіз архітектурних та структурних особливостей комп’ютерів) (2 курс) ОК Комп’ютерні системи (аналіз структури та організації обчислень, оцінка трудомісткості) (3 курс) ОК Комп’ютерна електроніка (аналіз параметрів схем на мікроелектронній базі) (2 курс)	<i>Структурно-декомпозиційні завдання</i> (спрямовані на розбиття складної системи/ схеми на базові елементи для розуміння їхньої ролі). <i>Аналітико-порівняльні завдання</i> (завдання на порівняння кількох технологічних рішень, структур або алгоритмів для виявлення їхніх переваг та недоліків). <i>Аналітико-діагностичні завдання</i> (спрямовані на аналіз симптомів, пошук вузьких місць «bottlenecks», або локалізацію несправностей у системі/кодів). <i>Завдання з критеріального оцінювання / експертизи</i> (завдання на аналіз ефективності, швидкості чи надійності за заданими критеріями). <i>Завдання на причинно-наслідковий аналіз</i> (виявлення того, як зміна одного параметра впливає на поведінку всієї системи).
Синтез – це уявне об’єднання частин, властивостей, дій в єдине ціле.	
ОК Комп’ютерна логіка (синтез комбінаційних логічних схем та цифрових керуючих автоматів) (1-2 курс) ОК Комп’ютерна схемотехніка (синтез типових функціональних вузлів комп’ютерів та операційних автоматів) (2 курс) ОК Об’єктно-орієнтоване програмування (синтез комплексних програмних рішень шляхом поєднання окремих класів та об’єктів). (2 курс)	<i>Проектно-конструкторські завдання</i> (завдання на створення нової апаратної схеми, мережевої топології або архітектури ПЗ з базових елементів на основі технічного завдання). <i>Оптимізаційно-синтетичні завдання</i> (спрямовані на конструювання системи в умовах жорстких обмежень. Наприклад, синтез схеми з мінімальною кількістю логічних вентилів або найменшою затримкою сигналу). <i>Інтеграційні завдання</i> (проблемні завдання на об’єднання кількох розрізнених підсистем у цілісний апаратно-програмний комплекс, наприклад, об’єднання операційного та керуючого автоматів). <i>Алгоритміко-модельовальні завдання</i> (завдання на побудову графа алгоритму або узагальненої моделі майбутнього пристрою/програми перед її фізичною чи програмною реалізацією).
Класифікація (від лат. <i>classis</i> – розряд, клас та <i>facio</i> – роблю, розкладаю) – мисленнєва операція, спрямована на розподіл предметів, явищ і понять за видами, ознаками, класами, групами, розрядами залежно від їх істотних ознак в групі, підгрупі, класі – кінцева або нескінченна сукупність виділених за деякою ознакою предметів, що розуміється як ціле)	
ОК Організація баз даних (класифікація моделей даних, нормальних форм, типів зв’язків між сутностями) (2 курс)	<i>Систематизаційно-структурні завдання</i> (проблемні задачі, де неструктурований набір даних або технічних вимог потрібно розбити на

ОК Захист інформації в комп'ютерних системах (класифікація загроз безпеці, криптографічних методів, моделей розмежування доступу) (4 курс) ОК Теорія інформації та кодування (класифікація завадостійких кодів, алгоритмів стиснення, типів завад у каналах зв'язку) (1 курс) ОК Паралельні та розподілені обчислення (класифікація багатопроцесорних архітектур за таксономією Флінна, типів паралелізму) (3 курс) ОК Пристрої зв'язку з об'єктом (класифікація датчиків, аналогоцифрових та цифроаналогових перетворювачів за ознаками) (3 курс) ОК. Електротехніка і електроніка (класифікація електронних компонентів, джерел живлення та електричних сигналів) (1 курс) ОК Технічні засоби передачі інформації (класифікація середовищ передачі, лінійних кодів та мережесих інтерфейсів) (2 курс)	логічні класи для подальшого опрацювання – наприклад, при проєктуванні реляційної бази даних). <i>Ідентифікаційно-категоризаційні завдання</i> (завдання на визначення приналежності невідомого об'єкта до певного класу на основі його симптомів чи характеристик; наприклад, ідентифікація типу мережевої атаки чи типу апаратного збою). <i>Завдання з критеріального підбору на основі класифікаторів</i> (завдання на вибір найкращого рішення з наявного класу пристроїв/алгоритмів під конкретні умови: наприклад, вибір завадостійкого коду під специфічний рівень шуму). <i>Таксономічні завдання</i> (спрямовані на самостійну розробку студентом багаторівневої класифікаційної схеми для певної технічної задачі).
Порівняння – мисленнєва операція, при якій встановлюються схожість і відмінності між об'єктами. Основою для порівняння, як правило, є істотні ознаки порівнюваних об'єктів.	
ОК Системне програмне забезпечення (порівняння архітектур ОС, файлових систем, засобів низькорівневого та високорівневого програмування) (3 курс) ОК Інформаційні технології (порівняння прикладного ПЗ, архітектурних рішень для побудови мереж, операційних систем) (1 курс) ВК Вступ до архітектури розумних пристроїв Інтернету речей (порівняння сенсорів, платформ IoT, протоколів бездротової передачі даних щодо енергоефективності) (3 курс) ВК Програмування в середовищі сучасних ОС (порівняння різних парадигм програмування: структурної, модульної, об'єктно-орієнтованої) (4 курс) ОК Навчальна практика з проєктування цифрових пристроїв (порівняння різних апаратних реалізацій одного алгоритму за швидкістю, апаратними витратами) (3 курс)	<i>Альтернативно-оцінювальні завдання</i> (завдання на обґрунтований вибір однієї з кількох існуючих технологій/схем для вирішення конкретної проблеми індустрії з урахуванням жорстких обмежень. Наприклад, вибір протоколу для IoT-пристрою з автономним живленням). <i>Контрастивно-аналітичні завдання</i> (спрямовані на виявлення глибинних відмінностей у зовні схожих об'єктах: наприклад, знайти відмінності в управлінні пам'яттю між Windows та Linux). <i>Експериментально-бенчмаркінгові завдання</i> (проблемні практичні завдання, де студент має реалізувати два різні алгоритми або схеми, заміряти їхні реальні показники швидкодії чи ресурсоемності, і на основі зіставлення зробити висновки). <i>Компаративно-еволюційні завдання</i> (порівняння попереднього та сучасного покоління технологій для виявлення тенденцій та розуміння причин еволюції апаратних чи програмних рішень).
Узагальнення – мисленнєва операція, при якій відбувається отримання висновку на основі уявного об'єднання предметів і явищ за їх загальними та істотними ознаками	
ОК Об'єктно-орієнтоване програмування (узагальнення властивостей та методів об'єктів для створення базових класів та використання шаблонів) (2 курс)	<i>Завдання на алгоритмічне / структурне узагальнення (патерізацію)</i> (проблемні ситуації, де студент має написати не просто рішення (скрипт чи схему) для однієї задачі, а створити універ-

ОК Технології проектування цифрових систем (створення бібліотечних модулів для повторного використання у проектах) (2 курс) ОК Паралельні та розподілені обчислення (узагальнення класичних послідовних алгоритмів до універсальних патернів розпаралелювання) (3 курс) ОК Комп'ютерні системи (виведення загальних архітектурних принципів організації обчислень на основі аналізу різних сучасних комп'ютерних платформ) (3 курс) ОК Виробнича практика (узагальнення отриманого розрізненого практичного досвіду роботи з обладнанням/ПЗ до рівня професійних навичок та інструкцій).	сальний шаблон або параметризований модуль, здатний обробляти цілий клас подібних проблем за принципом DRY Don't Repeat Yourself). <i>Індуктивно-узагальнювальні (емпіричні) завдання</i> (завдання на виведення загальної закономірності на основі серії проведених експериментів; наприклад, після серії дослідів з мережевим трафіком чи навантаженням процесора самостійно сформулювати правило падіння продуктивності). <i>Трансферні (екстраполяційні) завдання</i> (завдання на перенесення узагальненого принципу або алгоритму, виявленого та засвоєного в одній системі, на абсолютно нову, ще не вивчену технологію чи платформу). <i>Завдання на концептуалізацію досвіду (трабл-шутинг завдання)</i> (проблемні задачі рефлексивного характеру, коли несистемні та хаотичні результати пошуку та усунення несправностей у кодї чи мережі потрібно узагальнити у вигляді єдиного алгоритму дій, інструкції або протоколу).
Абстрагування (від лат. <i>abstractio</i> – відволікання) – це мисленнєва операція, яка виділяє окремі ознаки, елементи та відділяє їх від інших і від самих об'єктів.	
ОК Дискретна математика (абстрагування від фізичної природи об'єктів для побудови графів, множин та логічних функцій) (1 курс) ОК Об'єктно-орієнтоване програмування (моделювання реальних систем через абстрактні класи, інтерфейси та віртуальні функції) (2 курс) ОК Організація баз даних (побудова інфо-логічної (ER) моделі даних, абстрагування від специфіки конкретної СУБД на етапі концептуального проєктування) (2 курс) ОК Архітектура комп'ютерів (абстракція рівня системи команд – ISA, яка відокремлює програмне забезпечення від апаратних деталей) (2-3 курс) ОК Комп'ютерна логіка (побудова абстрактних цифрових автоматів Мілі та Мура до етапу їх структурної та апаратної реалізації) (1-2 курс)	<i>Концептуально-моделювальні завдання</i> (проблемні задачі, де студенту дається неструктурований опис реальної чи бізнес ситуації, з якої він повинен виділити лише суттєві параметри та перетворити їх на абстрактну модель: наприклад, побудувати граф комп'ютерної мережі або ERдіаграму бази даних). <i>Завдання на об'єктно-орієнтоване абстрагування</i> (завдання на створення ієрархії класів ПЗ, де найвищий рівень є абстрактним, що визначає лише контракти/інтерфейси системи без реалізації технічних деталей). <i>Завдання на багаторівневу (архітектурну) абстракцію</i> (проблемні ситуації, де потрібно розв'язати задачу суворо в межах одного логічного рівня, ізолювавши його від інших: наприклад, налаштування маршрутизації на мережевому рівні, свідомо відволікаючись від того, яке фізичне середовище передачі використовується). <i>Ідеалізаційні завдання</i> (спрямовані на пошук принципового алгоритмічного рішення в ідеальних умовах. Наприклад, розробка графа переходів абстрактного автомата керування без урахування реальних фізичних затримок сигналів в апаратурі).
Дедукція (лат. <i>deductio</i>, від <i>deduco</i> – «низводжу, відводжу») виведення часткових наслідків із загальних передумов; застосування загальних законів, правил, стандартів або алгоритмів до конкретних інженерних ситуацій	

ОК Захист інформації в комп'ютерних системах (застосування загальних політик безпеки, наприклад ISO/IEC 27035 або концепції Cyber Kill Chain, до захисту конкретної інфраструктури). (4 курс) ОК Дискретна математика (застосування загальних законів та аксіом булевої алгебри для логічного виведення або мінімізації конкретної функції). (1 курс) ОК Електротехніка і електроніка (застосування фундаментальних законів Кірхгофа чи Ома для розрахунку параметрів струму в конкретному складному електричному колі). (1 курс)	<i>Дебагінгові (траблшутингові) завдання</i> (проблемні ситуації, де студент, спираючись на загальне знання архітектури, має логічно вирахувати та локалізувати конкретне місце збою в коді, мережі або схемі – мистецтво методологічного мислення при пошуку помилок). <i>Завдання на конкретизацію патернів / стандартів</i> (проблеми, де вимагається застосувати загальний архітектурний шаблон або протокол (наприклад, STP чи OSPF) для побудови рішення в умовах конкретного замовника). <i>Прогностично-дедуктивні завдання</i> (спрямовані на логічне передбачення: спираючись на загальні правила функціонування системи, студент має спрогнозувати, як вона поведе себе при введенні конкретних екстремальних, так званих "крайових", значень). <i>Експертно-діагностичні завдання з безпеки (СТГзавдання)</i> (застосування загальних принципів криптоаналізу чи реверсінжинірингу для експлуатації вразливості або перевірки надійності конкретного бінарного файлу чи вузла мережі).
Індукція (лат. <i>inductio</i> «привести (до)») метод пізнання, за якого загальний висновок про властивості множини елементів або явищ робиться на основі дослідження окремих (часткових) фактів, експериментів чи спостережень	
ОК Електротехніка і електроніка (виведення загальних емпіричних закономірностей та перевірка законів Кірхгофа/Ома на основі серії вимірювань струмів та напруг у різних вузлах). (1 курс) ОК Комп'ютерні системи (виявлення загальних причин «вузьких місць» продуктивності системи на основі серії окремих бенчмарків та стрестестів). (3 курс) ОК Паралельні та розподілені обчислення (виведення емпіричного правила меж розпаралелювання (закону Амдала) на основі замірів швидкодії для 2, 4, 8, 16 потоків). (3 курс) ОК Пристрої зв'язку з об'єктом (побудова узагальненої калібрувальної кривої або правила фільтрації шумів на базі масиву одиничних показів датчиків АЦП). (3 курс) ОК Комп'ютерна електроніка (встановлення загальної логічної функції невідомого мікроелектронного елемента шляхом подачі на вхід усіх можливих комбінацій сигналів). (2 курс) ВК Оптично волоконні мережі (формулювання загальних правил деградації оптич-	<i>Експериментально-дослідницькі завдання</i> (проблемні задачі, де студент має провести серію однотипних дослідів (наприклад, змінюючи розмір кешу або частоту процесора), щоб самостійно вивести та сформулювати загальну залежність). <i>Завдання типу «Чорний ящик» (реверс-інжиніринг)</i> (ситуації, коли алгоритм або схема приховані; студент подає різні вхідні дані, фіксує вихідні результати і шляхом індукції відтворює загальну внутрішню логіку роботи пристрою/програми). <i>Завдання на висування та перевірку емпіричних гіпотез</i> (проблемні ситуації, де на основі кількох зафіксованих симптомів (наприклад, періодична втрата пакетів у мережі) студент висуває узагальнену гіпотезу про причину збою, яку потім має експериментально підтвердити). <i>Статистично-аналітичні завдання</i> (завдання на збір великого масиву розрізнених даних (логів сервера, показників завантаженості пам'яті) та індуктивне зведення їх до єдиної моделі поведінки системи).

ного сигналу на основі аналізу серії різних рефлектограм). (4 курс)	
Формалізація відображення об'єктів, явищ або процесів у знаково-символічній формі – за допомогою математичних формул, логічних виразів, мов програмування чи спеціалізованих мов опису апаратури	
ОК Дискретна математика (формалізація логічних висловлювань у вигляді булевих функцій та графів). (1 курс) ОК Комп'ютерна логіка (формалізація словесного опису поведінки цифрового автомата у вигляді таблиць переходів/виходів та логічних рівнянь). (1-2 курс) ОК Об'єктно-орієнтоване програмування (формалізація властивостей та поведінки реальних сутностей за допомогою синтаксису мови програмування – класів, інтерфейсів, методів). (2 курс) ОК Технології проектування цифрових систем (формальний опис апаратних засобів за допомогою мов опису апаратури HDL). (2 курс) ОК Теорія інформації та кодування (формалізація процесів передачі даних та виправлення помилок за допомогою строгих математичних алгоритмів кодування). (1 курс)	<i>Трансляційно-символічні завдання</i> (проблемні ситуації, де неструктурований текстовий або словесний опис вимог замовника потрібно перекласти на строгу математичну, логічну або алгоритмічну мову, усунувши будь-яку неоднозначність). <i>Алгоритмічно-кодувальні завдання</i> (завдання на безпосереднє перетворення абстрактної моделі або ідеї, наприклад, структурної схеми чи UML діаграми, у синтаксично правильний програмний код). <i>Завдання з апаратної формалізації</i> (HDL-завдання) (інженерні проблеми, де очікувану фізичну поведінку майбутньої мікросхеми потрібно описати суворою мовою апаратного рівня (Hardware Description Language) для подальшого синтезу на ПЛІС/FPGA).
Моделювання вивчення об'єкта чи явища шляхом створення та дослідження його копії (моделі), яка заміщує оригінал і відображає його найважливіші властивості	
ОК Навчальна практика з проектування цифрових пристроїв (імітаційне моделювання розроблених операційних та керуючих автоматів перед їх апаратною реалізацією). (2 курс) ОК Організація баз даних (концептуальне, логічне та фізичне моделювання предметної області, побудова ER-моделей). (2 курс)	<i>Імітаційно-моделювальні завдання</i> (проблемні ситуації, де студент повинен створити віртуальну копію (імітаційну модель) комп'ютерної мережі або мікросхеми в симуляторі для безпечного тестування її роботи в різних режимах навантаження). <i>Завдання з концептуального (інформаційного) моделювання</i> (завдання, де неформалізовані вимоги предметної області потрібно перетворити на строгу структурну модель. Наприклад, розробити ER-діаграму для майбутньої бази даних або UML діаграму класів для ПЗ). <i>Прогностично-експериментальні завдання на базі моделей</i> (завдання дослідницького характеру, де студент має вивчити поведінку вже готової моделі, критично змінюючи її параметри, щоб спрогнозувати вузькі місця «bottlenecks» реальної системи). <i>Математико-алгоритмічні завдання</i> (завдання на побудову формальної математичної або логічної моделі процесу обчислень для доведення її коректності перед етапом безпосереднього кодування чи апаратного синтезу).

ПРИКЛАДИ КЕЙСІВ РІЗНОГО ТИПУ

НАУКОВО-ДОСЛІДНІ ТА АНАЛІТИЧНІ КЕЙСИ

ОК «Комп'ютерні мережі»

Назва кейсу **«Оптимізація стеку протоколів прикладного рівня для сенсорної мережі в умовах нестабільного зв'язку».**

Контекст (сюжет). Мережа розумних теплиць, розташованих у віддаленому регіоні, збирає телеметрію (температура, вологість, освітленість) з тисяч мікроконтролерів і передає їх на центральний сервер через стільникову мережу (EDGE/3G).

Проблемна ситуація. Поточна система використовує архітектуру REST API поверх HTTP. Через нестабільне покриття встановлення TCP-з'єднань часто обривається, відбувається повторна передача даних. Це призводить до надмірного використання трафіку (накладні витрати заголовків HTTP і TCP значно перевищують корисне навантаження), що швидко виснажує батареї автономних датчиків.

Інформаційна база: 1. Дампи мережевого трафіку (pcap-файли з Wireshark) поточної HTTP-комунікації. 2. RFC документація та специфікації протоколів MQTT та CoAP. 3. Статистика розряду акумуляторів датчиків за останній місяць.

Завдання:

1. Проаналізувати дампи трафіку та розрахувати відсоток накладних витрат (overhead) при використанні HTTP для передачі 50 байт корисних даних.
2. Провести порівняльний аналіз протоколів MQTT (на базі TCP) та CoAP (на базі UDP) для заданих умов.
3. Запропонувати та обґрунтувати новий стек протоколів, який мінімізує мережеві затримки та продовжить час автономної роботи пристроїв.

ОК «Організація баз даних»

Назва кейсу **«Оцінка доцільності міграції реляційної архітектури на NoSQL для зберігання часових рядів».**

Контекст (сюжет). Національна система екологічного моніторингу щосекунди збирає дані з понад 10 000 станцій контролю якості повітря. Дані зберігаються у централізованій реляційній базі даних PostgreSQL.

Проблемна ситуація. Зі збільшенням кількості станцій система почала деградувати. Операції запису (INSERT) створюють блокування таблиць, а аналітичні запити (наприклад, розрахунок середнього рівня CO₂ за місяць) виконуються хвилинами. Масштабування сервера (Scale-up) вичерпало свій ліміт. Команда розглядає перехід на спеціалізовані рішення для часових рядів (Time-Series DB) або Column-family бази даних (наприклад, Cassandra).

Інформаційна база. Поточна ER-діаграма реляційної бази. Плани виконання найважчих SQL-запитів (результати команди EXPLAIN ANALYZE). Метрики продуктивності дискової підсистеми (IOPS) та статистика читання/запису (співвідношення запису до читання – 98% до 2%).

Завдання:

1. Здійснити аналіз поточних «вузьких місць» (bottlenecks) реляційної моделі на основі наданих планів виконання запитів.
2. Дослідити архітектурні відмінності між класичними B-Tree індексами (PostgreSQL) та LSM-деревами (у NoSQL системах).
3. Оцінити системи за теоремою CAP і підготувати архітектурне рішення щодо доцільності (або недоцільності) міграції на InfluxDB чи Cassandra.

ОК «Захист інформації в комп'ютерних системах»

Назва кейсу **«Розслідування вразливостей віддаленого доступу та проєктування захищеної архітектури».**

Контекст (сюжет). Промислове підприємство швидко перевело частину інженерного персоналу на віддалену роботу, відкривши доступ до корпоративної мережі через RDP (Remote Desktop Protocol) з прокиданням портів на маршрутизаторі без використання VPN.

Проблемна ситуація. Аналітики відділу безпеки виявили аномальну активність: SIEM-система фіксує тисячі спроб авторизації в неробочий час із закордонних IP-адрес. Одночасно на кількох робочих станціях спостерігаються збої в роботі служби віддалених стільниць. Керівництво вимагає оцінити ризики та запропонувати системне рішення, яке забезпечить інфраструктуру, не блокуючи роботу співробітників.

Інформаційна база: 1. Витяги з логів міжмережевого екрана (Firewall) та SIEM-системи. 2. Офіційна технічна документація та звіти CVE щодо вразливостей протоколу RDP (наприклад, CVE-2019-0708 BlueKeep). 3. Поточна топологія корпоративної мережі.

Завдання:

1. На основі аналізу логів класифікувати тип атаки та описати потенційний вектор компрометації згідно з матрицею MITRE ATT&CK.
2. Провести дослідження технічної природи виявленої вразливості в архітектурі протоколу RDP.
3. Спроекувати та захистити концептуально нову архітектуру доступу (розглянути впровадження IPSec/OpenVPN, механізмів MFA або концепції Zero Trust Network Access), зваживши безпеку та зручність для користувачів.

ТЕХНІКО-ДІАГНОСТИЧНІ КЕЙСИ

ОК «Програмування в середовищі сучасних ОС»

Назва кейсу **«Відмова безсерверної функції (Cloud Function) через конфлікт середовища виконання».**

Контекст (сюжет). Команда розробників підтримує бекенд-інфраструктуру на базі GCP/AWS. Один із мікросервісів – Python-скрипт (Cloud Function), який автоматично парсить лог-файли з Cloud Storage, форматує їх та відправляє сповіщення адміністраторам.

Проблемна ситуація. Після чергового мінорного оновлення коду розробником-початківцем (додавання нових змінних середовища та оновлення логіки обробки), функція перестала розгортатися. Система CI/CD (Continuous

Integration) видає помилку "Deployment Failed", а в середовищі тестування скрипт раптово завершується з помилкою синтаксису, хоча локально у розробника нібито все працювало.

Інформаційна база: Додаток 1 (Трейсбек помилки виконання): File "main.py", line 42 def process_logs(event, context): IndentationError: unindent does not match any outer indentation level (або альтернативний лог) SyntaxError: invalid syntax (вказує на відсутність закриваючої фігурної дужки } в словнику конфігурації нижче по коду). Додаток 2 (Фрагмент Bash-скрипта експорту змінних середовища перед деплоєм): export API_KEY="a1b2"export ENV="prod"export REGION="eu-central-1"

Завдання для здобувача:

- 1.Провести аналіз наданих логів. Ідентифікувати одразу дві критичні технічні проблеми: помилку в синтаксисі Python-скрипта (наприклад, загублену фігурну дужку або порушення відступів) та семантичну помилку конкатенації рядків в Bash-скрипті (відсутність пробілів/розділювачів між командами export).

- 2.Розробити алгоритм валідації коду (Linting) та перевірки змінних середовища перед розгортанням.

- 3.Пояснити, як подібні синтаксичні помилки на рівні ОС та скриптових мов впливають на стабільність хмарних інфраструктур.

ОК «Комп'ютерні системи»

Назва кейсу **«Апаратна нестабільність та температурний тротлінг edge-сервера».**

Контекст (сюжет). На промисловому об'єкті розгорнуто edge-сервер (вбудована комп'ютерна система без активного охолодження) для управління роботизованою лінією. Система працює цілодобово.

Проблемна ситуація. З настанням літнього сезону система почала спонтанно перезавантажуватися в середині дня (між 13:00 та 15:00), коли навантаження на обчислювальне ядро максимальне. Програмних помилок у коді додатку не виявлено.

Інформаційна база: Додаток 1 (Витяг з системного журналу dmesg ОС Linux): [14:22:01] mce: [Hardware Error]: Machine check events logged [14:22:05] thermal thermal_zone0: critical temperature reached (98 C), shutting down [14:23:10] Under-voltage detected! (0x00050005). Додаток 2 (Графіки): Графік завантаження процесора (стабільно 85-90%) та графік напруги на шині живлення 5V (просідання до 4.6V під час пікових навантажень).

Завдання для здобувача:

1. На основі апаратних логів (dmesg) діагностувати комплексний характер проблеми (поєднання критичного перегріву thermal throttling та просідання напруги brownout).

2. Розробити алгоритм апаратної та програмної діагностики для інженера, який виїде на об'єкт (перевірка підсистеми живлення, радіаторів, налаштувань BIOS/UEFI щодо температурних лімітів).

3. Запропонувати системне рішення: як програмними методами (оптимізація планувальника ОС, зниження частоти undervolting/underclocking) стабілізувати роботу до заміни апаратної частини.

ОК «Захист інформації в комп'ютерних системах» (ЗІКС)

Назва кейсу **«Розслідування інциденту: Компрометація бекенд-системи через вразливість додатка».**

Контекст (сюжет). Компанія використовує ERP-систему (наприклад, Odoo 16), розгорнуту на базі ОС Linux. Доступ до неї має бути лише у співробітників, але сервер має "білу" IP-адресу.

Проблемна ситуація. Адміністратор баз даних помітив аномальне падіння продуктивності бази PostgreSQL. Одночасно система моніторингу зафіксувала різкий стрибок вихідного трафіку на невідомі закордонні IP-адреси.

Інформаційна база: Додаток 1 (Фрагмент логів веб-сервера Nginx / access.log): 192.168.1.100 - - [1/Jun/2024:02:15:30 +0300] "POST /web/login HTTP/1.1" 200 456 "-" "python-requests/2.25.1" (тисячі подібних запитів за хвилину). 203.0.113.55 - - [1/Jun/2024:02:20:10 +0300] "GET /web/content?model=ir.attachment&id=1&field=datas&download=true&filename=../etc/passwd HTTP/1.1" 200 ... Додаток 2 (Фрагмент auth.log): Успішний вхід по SSH користувача odoo з незнайомої IP-адреси відразу після подій у веб-логах.

Завдання для здобувача:

1. Діяти в ролі аналітика SOC (Security Operations Center). Проаналізувати логи та відновити хронологію атаки (Cyber Kill Chain). Який тип вразливості було експлуатовано (Brute-force, Path Traversal)?

2. Розробити невідкладний алгоритм локалізації інциденту (Incident Response Plan), щоб зупинити витік даних, не знищивши криміналістичні сліди в оперативній пам'яті сервера.

3. Сформуванати список заходів щодо захисту системи: налаштування WAF, зміна правил мережевого екрану (Network ACLs), впровадження політик стійких паролів та блокування перебору (на кшталт Fail2Ban).

ПРОЄКТНО-КОНСТРУКТОРСЬКІ КЕЙСИ

ОК «Комп'ютерні мережі»

Назва кейсу **«Проектування мережевої інфраструктури філії з жорстким бюджетом та інтеграцією Legacy-обладнання».**

Контекст (сюжет). Університет відкриває нову лабораторію та студентський коворкінг у старому корпусі. Виділений бюджет покриває лише закупівлю одного сучасного маршрутизатора ядра мережі.

Проблемна ситуація. Необхідно розгорнути мережу на 120 робочих місць (ПК), 30 IP-камер та забезпечити Wi-Fi покриття. В наявності є склад зі списаним, але робочим обладнанням (некеровані комутатори 100 Mbps та кілька керованих комутаторів L2 з 10-річним терміном експлуатації без підтримки PoE). Замовник вимагає жорсткої ізоляції трафіку (студенти не повинні мати доступу до IP-камер та мережі викладачів).

Інформаційна база: 1. План поверху із зазначенням довжини кабельних трас. 2. Перелік наявного застарілого обладнання на складі. 3. Прайс-лист на мідний кабель (Cat 5e/6), PoE-інжектори та розетки.

Завдання для здобувача:

1. Розробити логічну топологію мережі та схему VLAN для забезпечення ізоляції трафіку.
2. Скласти план IP-адресації (Subnetting) з використанням VLSM для оптимального розподілу адресного простору.
3. Створити високорівневу специфікацію обладнання: визначити, де саме в топології доцільно використати застарілі комутатори (щоб вкластися в бюджет), а як підключити IP-камери та точки доступу без підтримки PoE-комутаторів.

ОК «Архітектура комп'ютерів»

Назва кейсу **«Розробка архітектурної специфікації Edge-сервера для системи сортування продукції»**

Контекст (сюжет). Промислове підприємство модернізує конвеєр сортування. Планується встановити систему комп'ютерного зору, яка в реальному часі відбраковуватиме дефектні вироби.

Проблемна ситуація. Сервер має розміщуватися безпосередньо в цеху (Edge computing). Через високий рівень пилу в приміщенні використання систем активного охолодження (кулерів) заборонено. До того ж, габарити обчислювача жорстко обмежені розміром існуючої шафи автоматики (формат Mini-ITX).

Інформаційна база: 1. Вимоги алгоритму комп'ютерного зору (необхідна кількість TOPS – Trillion Operations Per Second, вимоги до пропускної здатності пам'яті). 2. Характеристики процесорів (x86, ARM), графічних прискорювачів та спеціалізованих NPU (TDP, габарити, інтерфейси).

Завдання для здобувача:

1. Проаналізувати вимоги алгоритму та обмеження середовища.
2. Синтезувати архітектуру системи: обрати комбінацію CPU + NPU або SoC (System-on-a-Chip), яка забезпечить потрібну продуктивність у межах теплового пакета (TDP), що допускає пасивне охолодження.
3. Скласти високорівневу специфікацію інтерфейсів введення-виведення (I/O) для підключення високошвидкісних промислових камер та застарілого ПЛК (програмованого логічного контролера), що керує скидачем конвеєра.

ОК «Комп'ютерні системи»

Назва кейсу **«Проектування мікросервісного шлюзу збору даних (IoT Gateway) з інтеграцією у хмару».**

Контекст (сюжет). Транспортна компанія встановлює на свої вантажівки телеметричні шлюзи для збору даних про стан двигуна та геолокацію. Зібрані дані мають відправлятися в хмарне середовище (наприклад, AWS або GCP) для аналітики.

Проблемна ситуація. Шлюз (Edge-пристрій на базі Linux) повинен збирати дані з бортового комп'ютера авто через CAN-шину та застарілі RS-232 датчики. Канал зв'язку (LTE/3G) в дорозі є вкрай нестабільним. Замовник вимагає, щоб архітектура програмного забезпечення шлюзу була модульною, гарантувала відсутність втрати даних при обриві зв'язку, а скрипти розгортання не містили хардкоду і легко переносилися на нові пристрої.

Інформаційна база: 1. Специфікація протоколів CAN та RS-232. 2. Документація на брокери повідомлень для вбудованих систем (наприклад, Mosquitto MQTT). 3. Довідник з написання bash-скриптів та автоматизації (export змінних середовища).

Завдання для здобувача:

1. Синтезувати архітектуру обміну даними всередині шлюзу: розробити логічну схему, де окремі Python-скрипти зчитують дані з RS-232/CAN і передають їх локальному брокеру повідомлень.
2. Запропонувати механізм буферизації (Store-and-Forward) на випадок втрати інтернет-з'єднання.
3. Написати фрагмент конфігураційного bash-скрипта для ініціалізації середовища (з правильним синтаксисом експорту змінних та без злиття команд), який автоматично запускатиметься при старті ОС Linux.

СИТУАЦІЙНО-СТРАТЕГІЧНІ КЕЙСИ

ОК «Системне програмне забезпечення»

Назва кейсу **«Оцінка ризиків та стратегія корпоративної міграції операційних систем»**.

Контекст (сюжет). Національна логістична компанія має парк із 5 000 робочих станцій, розподілених по 100 філіях. Усі комп'ютери працюють на базі ОС Windows 10, життєвий цикл якої наближається до кінця (End of Life).

Проблемна ситуація. ІТ-директорат стоїть перед вибором напрямку міграції. Перехід на Windows 11 вимагає заміни 40% апаратного забезпечення через відсутність апаратної підтримки TPM 2.0 на старих материнських платах, плюс закупівля нових корпоративних ліцензій. Альтернатива – переведення компанії на безкоштовну ОС на базі ядра Linux (наприклад, Ubuntu). Це рятує бюджет на «залізо» і ліцензії, але стара корпоративна ERP-система має лише клієнт під Windows.

Інформаційна база: 1. Звіт інвентаризації (Hardware Inventory), який показує кількість ПК, що не відповідають системним вимогам Windows 11. 2. Прайс-лист на ліцензії MS Windows Enterprise та вартість нового ПК бюджетного рівня. 3. Оцінка часу простою філій при зміні ОС та вартості перенавчання персоналу.

Завдання для здобувача:

1. Провести фінансовий порівняльний аналіз обох сценаріїв міграції.
2. Запропонувати стратегію розв'язання проблеми несумісності ERP-системи при виборі Linux (наприклад, розгортання термінальних

серверів (RDP), використання Wine або фінансування переписування ERP на вебклієнт).

3. Розробити поетапний план міграції (Rollout plan), який мінімізує вплив на безперервність бізнес-процесів (Business Continuity).

ОК «Організація баз даних»

Назва кейсу **«Стратегія масштабування архітектури баз даних для швидкозростаючого E-commerce проєкту».**

Контекст (сюжет). Успішний стартап у сфері електронної комерції готується до масштабної рекламної кампанії в "Чорну п'ятницю". Поточна монолітна реляційна база даних (PostgreSQL) працює на одному потужному сервері.

Проблемна ситуація. Під час навантажувального тестування виявлено, що база даних не витримає очікуваного потоку транзакцій. Архітектори пропонують два стратегічні шляхи:

Вертикальне масштабування (Scale-Up): купівля надпотужного корпоративного сервера (з більшою кількістю ядер та величезним обсягом RAM). Швидко впроваджується, не потребує переписування коду додатку, але має жорстку фізичну межу і космічну вартість апаратного забезпечення.

Горизонтальне масштабування (Scale-Out): перехід на розподілену архітектуру, впровадження шардингу (Sharding) бази даних або міграція частини некритичних даних (наприклад, сесій користувачів чи кошиків) у NoSQL кластер. Це дешевше за апаратним забезпеченням, не має меж масштабування, але потребує зупинки розробки нових функцій на 2 місяці для рефакторингу коду.

Інформаційна база: 1. Прогнози зростання транзакційного навантаження від відділу маркетингу. 2. Прайс-листи на сервери класу Hi-End та сервери середнього рівня (Commodity hardware). 3. Оцінка вартості робочого часу команди розробників (Developer hours) для реалізації рефакторингу під шардинг.

Завдання для здобувача:

1. Оцінити економічну ефективність (ROI) вертикального та горизонтального масштабування у короткостроковій та довгостроковій перспективі.
2. Визначити бізнес-ризики (що буде, якщо рефакторинг не встигнуть закінчити до "Чорної п'ятниці"?).
3. Запропонувати оптимальне управлінське рішення, обґрунтувавши компроміс між вартістю інфраструктури та часом виходу на ринок (Time-to-Market).

ОК «Захист інформації в комп'ютерних системах» (ЗІКС)

Назва кейсу **«Економіка кібербезпеки. Інвестиції в систему захисту проти потенційних збитків від інцидентів».**

Контекст (сюжет). Рада директорів середньої ІТ-компанії затверджує бюджет на наступний рік. Керівник служби інформаційної безпеки (CISO) наполягає на впровадженні дорогої системи EDR (Endpoint Detection and Response) та сертифікації за міжнародним стандартом безпеки.

Проблемна ситуація. Фінансовий директор (CFO) відмовляється погоджувати бюджет, аргументуючи це тим, що "нас ніколи не зламували", а впровадження нових політик уповільнить роботу розробників. Однак місяць тому компанію-конкурента було атаковано програмою-вимагачем (Ransomware), що призвело до зупинки їхньої роботи на тиждень.

Інформаційна база: 1. Вартість впровадження та підтримки EDR-рішення, а також аудиту інфраструктури. 2. Аналітичні дані (статистика) щодо середнього розміру викупу (ransom payout) при атаках вірусів-шифрувальників у цій галузі. 3. Розрахована вартість однієї години простою бізнесу компанії.

Завдання для здобувача:

1. Використовуючи методологію оцінки ризиків (наприклад, розрахунок ALE - Annualized Loss Expectancy), кількісно обґрунтувати економічну доцільність інвестицій у кібербезпеку.
2. Проаналізувати непрямі збитки від потенційного інциденту (репутаційні втрати, штрафи регуляторів, відтік клієнтів).
3. Сформулювати стратегічний план впровадження безпекових рішень, який би був компромісом між запитами CISO та можливостями бюджету CFO.

КЕЙСИ «РЕТРОСПЕКТИВНОГО АНАЛІЗУ»

БК «Програмування в середовищі сучасних ОС»

Назва кейсу «**Ціна синтаксичної помилки в CLI: Аналіз падіння інфраструктури AWS S3 (2017)**».

Контекст (сюжет). 28 лютого 2017 року регіон US-East-1 хмарного провайдера Amazon Web Services зазнав масштабного збою. Тисячі популярних вебсайтів, IoT-систем та бекенд-додатків припинили роботу, оскільки втратили доступ до об'єктного сховища S3.

Проблемна ситуація. Команда інженерів виконувала налагодження проблеми з білінговою системою. Інженер мав виконати Python-скрипт через AWS CLI для примусового перезавантаження невеликої кількості серверів. Однак через хибний синтаксис передачі аргументів у скрипт (відсутність коректних розділювачів або помилка в конкатенації рядків) команда була інтерпретована неправильно. Замість 2 серверів, скрипт відправив команду на видалення масштабної підсистеми серверів, що підтримували метадані всього регіону.

Інформаційна база: 1. Фрагмент документації до інструменту командного рядка (CLI) та логіки парсингу аргументів у Python (argparse). 2. Офіційний Post-Mortem звіт від AWS.

Завдання для здобувача:

1. Проаналізувати механізм передачі аргументів командного рядка в середовищі Linux. Як дрібна синтаксична помилка (наприклад, зайвий пробіл або відсутність лапок при експорті змінних) може радикально змінити семантику команди?
2. Розробити концепцію «запобіжників» (Guardrails) у коді скрипта розгортання, яка б фізично не дозволила видалити більше ніж 5% серверів за одну ітерацію, навіть якщо вхідні параметри задані хибно.

3. Запропонувати процедуру безпечного тестування CLI-інструментів (Dry-Run mode) перед їх застосуванням у production-середовищі.

ОК «Захист інформації в комп'ютерних системах» (ЗІКС)

Назва кейсу **«Вразливість інфраструктури до бічного переміщення: Аналіз епідемії NotPetya (2017)»**.

Контекст (сюжет). У червні 2017 року світ вразила найдорожча кібератака в історії (збитки понад 10 млрд доларів). Атака почалася в Україні через компрометацію легального оновлення бухгалтерського ПЗ М.Е.Дос, після чого поширилася на глобальні корпорації (Maersk, FedEx, Merck).

Проблемна ситуація. Шкідливе ПЗ використовувало не лише класичний фішинг, а й експлоїт EternalBlue (вразливість протоколу SMBv1 у Windows) та інструменти витягування паролів із пам'яті (Mimikatz). Як тільки один комп'ютер у мережі заражався, вірус автоматично та легітимно (з точки зору ОС) поширювався на всі інші пристрої в домені Active Directory, знищуючи MBR (Master Boot Record) жорстких дисків. Антивіруси виявилися безсилими.

Інформаційна база: 1. Опис вразливості CVE-2017-0144 (EternalBlue). 2. Матриця MITRE ATT&CK (розділи Lateral Movement та Impact). 3. Базова топологія несеgmentованої корпоративної мережі.

Завдання для здобувача:

1. Використовуючи методологію Cyber Kill Chain, розібрати етап Lateral Movement (бічне переміщення). Чому атака типу Supply Chain (через оновлення легального ПЗ) дозволила обійти зовнішні периметри захисту (Firewalls)?
2. Провести реверс-аналіз логіки поширення: як саме використання застарілого протоколу SMBv1 у поєднанні з архітектурою Windows дозволило експлоїту виконувати код від імені системи (SYSTEM).
3. Розробити стратегію нульової довіри (Zero Trust) та сегментації мережі, яка б ізолювала подібний інцидент у межах одного VLAN, не давши йому знищити всю інфраструктуру.

ОК «Архітектура комп'ютерів» (Частина 1)

Назва кейсу **«Апаратне переповнення та катастрофа: Аналіз аварії ракети-носія Ariane 5 (1996)»**.

Контекст (сюжет). 4 червня 1996 року під час свого першого польоту ракета Ariane 5, розробка якої коштувала 7 мільярдів доларів, вибухнула на 37-й секунді після старту.

Проблемна ситуація. Розслідування показало, що аварія сталася через програмно-апаратну помилку в Інерціальній навігаційній системі (SRI). Програмне забезпечення, успішно перенесене з попередньої моделі ракети (Ariane 4), намагалося перетворити 64-бітне число з плаваючою комою (яке відповідало за горизонтальну швидкість) у 16-бітне ціле число зі знаком. Оскільки Ariane 5 була значно швидшою, значення швидкості перевищило максимальну місткість 16-бітного регістра (32767). Виникло апаратне виключення «Переповнення операнда», яке не оброблялося програмним кодом, що призвело до повного відключення навігаційного комп'ютера.

Інформаційна база: 1. Специфікація представлення даних у пам'яті комп'ютера (стандарт IEEE 754 для чисел із плаваючою комою та додатковий код для цілих чисел). 2. Уривки зі звіту комісії з розслідування ESA (Європейського космічного агентства).

Завдання для здобувача:

1. На рівні архітектури процесора та регістрів змодельовати операцію перетворення 64-бітного Float у 16-бітний Integer. Довести математично, чому виникло апаратне переривання.
2. Проаналізувати помилку методології тестування: чому системні інженери вирішили не додавати блок обробки виключень (try-catch) для цієї конкретної змінної (підказка: економія ресурсів процесора в системах реального часу).
3. Запропонувати архітектурне рішення (як програмне, так і апаратне резервування), яке б дозволило бортовому комп'ютеру продовжити роботу навіть у разі критичної помилки в одному з математичних співпроцесорів.

МІЖДИСЦИПЛІНАРНІ КЕЙСИ

ОК «Програмування в середовищі сучасних ОС» + «Об'єктно-орієнтоване програмування»

Назва кейсу **«Інтеграція сучасного ПЗ зі «старим» обладнанням (Legacy Hardware) у лабораторії».**

Контекст (сюжет). Університетська лабораторія має дорогу вимірювальну установку (наприклад, спектрометр), яка працює лише через інтерфейс RS-232 під Windows XP.

Проблемна ситуація. Технічний виклик: потрібно інтегрувати дані з цього приладу в сучасну веб-систему (Python, Cloud) для дистанційного керування здобувачами. Адміністративний конфлікт: ІТ-відділ забороняє тримати Windows XP у мережі через вразливості, але замінити сам спектрометр за \$50,000 немає можливості.

Завдання: розробити «шлюз-ізолятор» (на базі Raspberry Pi або мікроконтролера), який підключається до старого спектрометра, зчитує дані через старий протокол, конвертує їх у сучасний JSON/MQTT і відправляє у хмару, повністю ізолюючи застарілий комп'ютер від корпоративної мережі.

ОК «Захист інформації в комп'ютерних системах» (ЗІКС) + «Правове забезпечення»

Назва кейсу **«Впровадження системи відеоспостереження з аналітикою в університетському містечку».**

Контекст. Керівництво університету вирішує встановити систему камер з розпізнаванням облич для забезпечення безпеки студентів та викладачів.

Проблемна ситуація. Технічний виклик: вибір архітектури (локальний сервер для обробки відео vs хмарне рішення), забезпечення конфіденційності даних (GDPR/Закон України «Про захист персональних даних»), розробка алгоритму, який мінімізує кількість хибних спрацювань.

Юридичний/Етичний конфлікт. Як отримати згоду на обробку біометричних даних від усіх студентів? Де фізично зберігати ці дані? Хто має до них доступ і як уникнути «стеження» без згоди?

Завдання: здобувач має спроектувати систему так, щоб вона була технічно ефективною, але юридично бездоганною (наприклад, реалізувати анонімізацію даних на рівні Edge-пристрою, де обличчя перетворюються на унікальні хеші без збереження фотографій).

КЕЙСИ «РЕАКТИВНОГО (ШВИДКОГО) РЕАГУВАННЯ»

ОК «Захист інформації в комп'ютерних системах» (ЗІКС)

Назва кейсу **«Виявлення спроби витоку даних (DLP-інцидент)»**.

Контекст (сюжет). Система запобігання витоку даних (DLP) надіслала критичне сповіщення: зі станції бухгалтера о 09:15 почалася масова передача файлів на зовнішній FTP-сервер.

Проблемна ситуація (Час на реакцію – 5 хвилин): Якщо ви заблокуєте мережу бухгалтера, зупиниться робота цілого відділу. Якщо ви проігноруєте – компанія може втратити конфіденційні звіти.

Інформаційна база: 1. Лог DLP: File access: "Report_2025.xlsx" accessed by user "acc_nadiya". Target: 185.xxx.xxx.xxx (External IP). 2. Активність користувача: Бухгалтер перебуває у відпустці (дані HR-системи).

Завдання:

1. Ізолювати вузол чи лише заблокувати вихідний трафік для конкретного IP-адресата?
2. Як мінімізувати збитки та зберегти доказову базу для розслідування?
3. Діяти згідно з алгоритмом Incident Response (Containment phase).

ОК «Комп'ютерні мережі»

Назва кейсу **«Петля комутації (Broadcast Storm) у сегменті мережі»**.

Контекст (сюжет). Користувачі цілого крила офісу раптово втратили доступ до мережі. Мережевий комутатор рівня доступу видає помилку CPUHOG та перезавантажується.

Проблемна ситуація (Час на реакцію – 10 хвилин). Мережа «лежить». Неможливо отримати доступ до інтерфейсу керування через SSH через перевантаження CPU.

Інформаційна база: 1. Звіт SNMP-моніторингу: Port Gi0/12 activity: 99% usage (Broadcast packets: 98% of total traffic). 2. Статус комутатора: Індикатори на портах безперервно блимають синхронно.

Завдання:

1. Локалізувати причину (ймовірна петля через неавторизований Wi-Fi роутер користувача).
2. Розробити план дій: відключити підозрілий порт фізично чи через консольний кабель (якщо доступний), щоб відновити сервіс для 40+ користувачів.
3. Запропонувати метод запобігання (наприклад, portfast, bpduguard).

ПРИКЛАДИ ЗАВДАНЬ ЩОДО ПРОБЛЕМАТИЗАЦІЇ КОМП'ЮТЕРНО-ІНЖЕНЕРНОЇ ДІЙСНОСТІ

Пакет проблемних завдань «Архітектурно-структурні суперечності сучасних обчислювальних систем»

ОК «Архітектура комп'ютерів»

Завдання 1. Конфлікт конвеєризації «Глибина проти Залежностей»

Проблемна ситуація. Команда розробників проєктує новий суперскалярний процесор і для підвищення тактової частоти пропонує збільшити довжину конвеєра з 14 до 24 стадій. Теоретичні розрахунки показують приріст пікової продуктивності на 40%. Однак під час симуляції на реальному програмному коді (бази даних з великою кількістю умовних переходів) реальна продуктивність системи падає на 15% порівняно з базовою 14-стадійною моделлю.

Суперечність: між прагненням до максимального апаратного паралелізму (шляхом подрібнення стадій) та інформаційними залежностями реального коду, що руйнують цей паралелізм.

Завдання здобувачу:

1. Аналітично локалізувати причину падіння продуктивності (проблематизувати вплив хибних прогнозів розгалужень та гранулярності конвеєра).
2. Запропонувати архітектурний компроміс: які додаткові апаратні блоки (наприклад, динамічне передбачення переходів, перейменування регістрів) необхідно інтегрувати, щоб зберегти 24-стадійний конвеєр без втрати реальної швидкодії.
3. Оцінити, як ці нововведення вплинуть на загальну надійність та тепловиділення кристала.

Методологічний акцент. Розвиток системного мислення через усвідомлення того, що оптимізація однієї метрики (частоти) без врахування контексту (типу програмного навантаження) призводить до деградації системи.

Завдання 2. Когерентність кешу: «Локальна швидкість проти Глобальної синхронізації»

Проблемна ситуація. У 64-ядерній серверній системі зі спільною пам'яттю спостерігається каскадне падіння пропускної здатності системної шини. Кожне ядро має великий L1/L2 кеш для прискорення локальних обчислень. Проте при спробі паралельної обробки єдиного масиву даних (наприклад, матричні обчислення), системна шина повністю блокується службовим трафіком, а ядра простоюють в очікуванні даних.

Суперечність: між необхідністю глибокого локального кешування (для мінімізації часу доступу) та вимогами строгої консистентності даних у розподіленому середовищі.

Завдання здобувачу:

1. Проаналізувати поведінку протоколу когерентності (наприклад, MESI) у багатоядерному середовищі під час інтенсивного запису спільних даних.
2. Проблематизувати поточну топологію спільної шини (Single Point of Failure для пропускної здатності).
3. Розробити та обґрунтувати альтернативну структурну організацію підсистеми пам'яті (наприклад, перехід до архітектури NUMA або впровадження каталог-орієнтованої когерентності), довівши її масштабованість.

Методологічний акцент. Формування здатності до декомпозиції складних монолітних архітектур та розуміння природи «вузьких місць» (bottlenecks) у багатопроекторних системах.

Завдання 3. Енерго-термальний бар'єр: «Dark Silicon (Темний кремній)»

Проблемна ситуація. Закон Мура дозволив інженерам розмістити на кристалі вдвічі більше транзисторів, зберігши колишню площу. Спроековано 16-ядерний мобільний процесор. Однак тепловий пакет (TDP) мобільного пристрою жорстко лімітований 5

Ватами. При спробі подати живлення на всі 16 ядер одночасно на максимальній частоті, кристал критично перегрівається за мілісекунди.

Суперечність: між фізичною можливістю розмістити колосальну кількість обчислювальних елементів та тепловими/енергетичними лімітами, що забороняють їх одночасне використання.

Завдання здобувачу:

1. Дослідити феномен «темного кремнію» та проблематизувати екстенсивний шлях розвитку архітектури.

2. Спроекувати стратегію динамічного управління енергоспоживанням (гетерогенна архітектура типу ARM big.LITTLE, clock gating, power gating).

3. Сформулювати правила (політики) для планувальника завдань операційної системи: як ефективно розподіляти потоки, враховуючи наявність «темного кремнію», щоб не порушити тепловий баланс.

Методологічний акцент. Навчання роботі з ресурсно-технологічними лімітами та розв'язання задач в умовах жорстких фізичних обмежень.

Завдання 4. Архітектурний борг: «Парадокс розширення системи команд»

Проблемна ситуація: Процесорна архітектура x86 еволюціонує десятиліттями, зберігаючи зворотну сумісність. Для прискорення нових задач (графіка, ШІ, шифрування) інженери постійно додають нові складні інструкції (AVX, AES-NI тощо). Зараз декодер команд процесора займає понад 15% площі ядра і споживає непропорційно багато енергії, при цьому старі (легасі) інструкції майже не використовуються сучасним софтом, але апаратно підтримуються.

Суперечність: Між необхідністю забезпечення інтеграційної сумісності з попередніми поколіннями ПЗ (легасі) та стрімким накопиченням апаратного «технологічного боргу», що знижує загальну енергоефективність ядра.

Завдання здобувачу:

1. Здійснити аудит архітектурного боргу платформи з набором складних інструкцій (CISC).

2. Запропонувати структурне рішення, яке дозволить процесору швидко обробляти старий код, не перевантажуючи фізичний дизайн нових виконавчих блоків (проблематизація переходу до внутрішніх RISC-мікрооперацій).

3. Оцінити ризики та наслідки повної відмови від зворотної сумісності (створення «чистої» архітектури з нуля).

Методологічний акцент. Формування навичок оцінки технологічного боргу та розуміння інженерних компромісів між підтримкою легасі-систем та еволюційним розвитком платформи.

Пакет проблемних завдань

«Ресурсно-технологічні ліміти комп'ютерних систем»

ОК «Комп'ютерні системи»

Завдання 1. Ілюзія пікової продуктивності: «Пляшкова горлянка фон Неймана»

Тема заняття: Лабораторна робота 2. Визначення швидкодії EOM.

Проблемна ситуація: Згідно з синтетичними тестами (бенчмарками), процесор демонструє теоретичну швидкодію у 150 GFLOPS. Проте під час запуску реальної задачі з обробки великої матриці зображень (4K відеострім) реальна швидкодія падає до 12 GFLOPS. Утилізація ядра становить 100%, але корисна робота майже не виконується — процесор простоє.

Суперечність: Між колосальною швидкістю обчислювального ядра та фізичними обмеженнями пропускної здатності шини оперативної пам'яті (Memory Wall).

Завдання здобувачу:

1. Локалізувати проблему через аналіз відсотка промахів у кеш-пам'яті (Cache Miss Rate).

2. Застосувати ресурсно-технологічний компроміс: переписати код (наприклад, використати блочну обробку матриць – Loop Tiling), щоб алгоритм працював виключно в межах об'єму L1/L2 кешу.

Методологічний акцент. Формування розуміння того, що реальна швидкодія системи лімітується не частотою процесора, а швидкістю доставки даних до нього.

Завдання 2. Апаратний ліміт розрядності: «Криза переповнення»

Тема заняття: Лабораторна робота 5. Робота арифметико-логічного пристрою центрального процесора.

Проблемна ситуація: Під час моделювання роботи АЛП для криптографічного застосунку розробляється модуль генерації великих простих чисел. На певному етапі алгоритм починає видавати некоректні від'ємні значення, хоча логіка коду є абсолютно правильною. Жодних повідомлень про помилку компілятор не видає (silent error).

Суперечність: між математичною вимогою обчислення нескінченно великих чисел та жорстким фізичним лімітом розрядності регістрів АЛП (64 біти).

Завдання здобувачу:

1. Проблематизувати апаратне обмеження цілочисельної арифметики та виявити точку цілочисельного переповнення (Integer Overflow).

2. Розробити концепцію програмного подолання цього фізичного ліміту (наприклад, реалізація довгої арифметики / BigInt), аргументувавши, як це вплине на загальний час виконання операцій (trade-off між точністю та швидкістю).

Методологічний акцент. Розвиток умінь бачити приховані загрози апаратних лімітів, які не відловлюються стандартними засобами зневадження (дебагінгу).

Завдання 3. Ціна комунікації: «Парадокс розпаралелювання»

Тема заняття (за РНП): Лабораторна робота 6. Знайомство із системою Paralab.

Проблемна ситуація: Складне математичне моделювання на одному ядрі триває 100 секунд. Інженер-початківець вирішує прискорити процес у 100 разів і запускає завдання на 100 ядрах суперкомп'ютера. Всупереч очікуванням, час виконання зростає до 120 секунд.

Суперечність: між наявністю гігантського пулу обчислювальних ресурсів та лімітами мережевої інфраструктури, що витрачає більше часу на синхронізацію процесів, ніж на самі обчислення.

Завдання здобувачу:

1. Застосувати закон Амдала (Amdahl's Law) для аналітичного доведення того, чому додавання ядер призвело до деградації системи.

2. Проблематизувати співвідношення часу обчислень до часу комунікації. Змінити гранулярність паралельного алгоритму (зробити блоки даних більшими), щоб мінімізувати пересилання службових повідомлень між вузлами.

Методологічний акцент. Формування критичного ставлення до екстенсивного масштабування та розуміння фізичної ціни синхронізації у багатопотокових середовищах.

Завдання 4. Межі лінійного масштабування: «Вузьке місце архітектури»

Тема заняття: Лабораторна робота 7. Вивчення кластерних структур.

Проблемна ситуація. Веб-сервіс розміщено на кластері з 3-х серверів. Він стабільно обробляє 10 000 запитів на секунду. Під час рекламної кампанії навантаження зростає вдесятеро. Адміністратор динамічно розширює кластер до 30 серверів (горизонтальне масштабування), але замість збільшення пропускної здатності вся система завмирає, оскільки єдина центральна база даних блокує всі транзакції через фізичну межу швидкості запису на диск (IOPS).

Суперечність: між можливістю безмежного горизонтального масштабування обчислювальних вузлів (web-серверів) та жорстким фізичним лімітом вводу/виводу єдиної спільної точки доступу до даних (бази даних).

Завдання здобувачу:

1. Діагностувати проблему вичерпання фізичного ліміту IOPS на головному вузлі кластера.

2. Розробити нову архітектурну стратегію подолання цього ліміту (наприклад, шардування бази даних, запровадження in-memoery кешування типу Redis, або асинхронні черги повідомлень).

Методологічний акцент. Системне мислення, що виводить інженера за межі налаштування окремого вузла до розуміння глобальних дефіцитів ресурсів в умовах кластерної архітектури.

Пакет проблемних завдань

«Інтеграційна та еволюційна сумісність мережевих систем»

ОК «Комп'ютерні мережі»

Завдання 1. Пізнавальна суперечність «Пастка автоузгодження»

Тема заняття: Лабораторна робота 4. Під'єднання фізичного рівня.

Проблемна ситуація: Під час модернізації серверної кімнати до старого промислового комутатора (який керує критичним конвеєрним обладнанням і не може бути замінений) підключили новий гігабітний сервер. Лінк (фізичний зв'язок) піднявся успішно, індикатори світяться. Проте швидкість передачі даних є катастрофічно низькою, а в логах фіксується величезна кількість колізій (late collisions) і втрачених кадрів.

Суперечність: між сучасними алгоритмами автоузгодження (Auto-Negotiation) швидкості та застарілою логікою роботи легасі-обладнання, яке не підтримує повнодуплексний режим (Full-Duplex).

Завдання здобувачу:

1. Локалізувати проблему на фізичному рівні (діагностувати Duplex Mismatch).

2. Проблематизувати ситуацію: чому сучасний стандарт інтеграції "plug-and-play" призвів до деградації мережі?

Запропонувати та обґрунтувати ручне налаштування параметрів інтерфейсу (швидкості та дуплексу) як метод зворотної сумісності зі старим обладнанням.

Методологічний акцент. Руйнування ілюзії «універсальної сумісності» сучасних технологій і формування розуміння того, що новітнє обладнання вимагає свідомої деградації власних можливостей для роботи з легасі-системами.

Завдання 2. Прихована фрагментація «Конфлікт інкапсуляції»

Тема заняття: Лабораторна робота 3. Дослідження моделей TCP/IP та OSI. Wireshark.

Проблемна ситуація. Компанія впровадила нову систему шифрування трафіку (VPN) між двома філіями. До цього всі бази даних працювали ідеально. Після інтеграції VPN малі пакети даних передаються миттєво (наприклад, ping проходить), але спроба завантажити великий звіт через HTTP або передати файл призводить до нескінченного зависання сесії ("чорна діра" маршрутизації).

Суперечність: між розміром корисного навантаження, яке формує прикладний рівень легасі-системи, та зміненими лімітами каналного/мережевого рівня (зменшення MTU через додаткові заголовки інкапсуляції VPN).

Завдання здобувачу:

1. За допомогою аналізатора Wireshark виявити, на якому рівні моделі OSI виникає скидання пакетів (Drop).

2. Проблематизувати механізм фрагментації: чому прапорець "Don't Fragment" (DF) у заголовку IPv4, який генерує стара система, блокує еволюційний перехід на захищені тунелі.

3. Вирішити інтеграційний конфлікт шляхом коригування розміру сегмента (MSS Clamping) на маршрутизаторі.

Методологічний акцент. Розвиток уміння бачити систему наскрізно (через усі рівні моделі OSI) та розуміння ціни "нашарування" нових протоколів поверх існуючої інфраструктури.

Завдання 3. Криза злиття «Топологічний колапс»

Тема заняття : Лабораторна робота 8. Обчислення підмереж IPv4.

Проблемна ситуація. Відбулося злиття двох компаній. Інженеру доручено об'єднати їхні локальні мережі. Проте аудит показує, що обидві компанії історично використовують один і той самий легасі-діапазон приватних адрес: 192.168.1.0/24. Пряме фізичне з'єднання призведе до глобального конфлікту IP-адрес. Зупинити роботу будь-якої з мереж для повної переадресації бізнес забороняє.

Суперечність: між вимогою фізичної та логічної інтеграції двох гетерогенних систем та історично сформованою несумісністю їхніх адресних просторів.

Завдання здобувачу:

1. Проблематизувати наслідки дублювання IP-адрес на маршрутизацію трафіку.
2. Розробити еволюційний план гібридної інтеграції: застосувати технологію трансляції адрес (NAT/Double NAT) на стику двох мереж як тимчасове рішення.
3. Спроекувати нову, масштабовану ієрархію підмереж IPv4 (VLSM) для поступової безшовної міграції обох філій у майбутньому.

Методологічний акцент. Формування навичок вирішення архітектурних конфліктів у реальному бізнес-середовищі, де технічно правильне рішення (переналаштувати все з нуля) є неможливим через вимоги безперервності процесів.

Завдання 4. Конфлікт довіри: «Отруєння легасі-середовища»

Тема заняття: Лабораторна робота 7. Визначення MAC-адрес та IP-адрес Дослідження ARP-таблиці.

Проблемна ситуація. У старій "плоскій" мережі компанії, яка не має поділу на віртуальні підмережі (VLAN), з'явився новий смарт-термінал. Раптом весь трафік, призначений для головного сервера, почав зникати. Дослідження показало, що новий пристрій постійно розсилає широкомовні ARP-запити, які перезаписують таблиці на всіх комутаторах, перенаправляючи трафік сервера на себе.

Суперечність: між базовою, історично закладеною "довірою" протоколу ARP до будь-яких відповідей у локальному сегменті та сучасними вимогами до безпеки в гетерогенних середовищах.

Завдання здобувачу:

1. Дослідити механізм підміни (ARP Spoofing) на рівні логіки оновлення таблиць.
2. Проблематизувати архітектуру легасі-мережі: чому один неавторизований (або некоректно налаштований) пристрій здатен зруйнувати логіку взаємодії цілого підприємства.
3. Запропонувати методи "еволюційного лікування": від статичних ARP-записів на критичних вузлах до інтеграції сучасних технологій типу Dynamic ARP Inspection (DAI).

Методологічний акцент. Зміна парадигми мислення від «мережа працює, якщо є лінк» до «інтеграція нового вузла створює потенційну вразливість для старої архітектури».

Пакет проблемних завдань

«Інженерна безпека та стійкість мережевих систем»

ОК «Комп'ютерні мережі», Частина 2

Завдання 1. Зброя з резервування: «Атака на кореневий міст (STP Spoofing)»

Тема заняття: Резервування мереж.

Проблемна ситуація. Для забезпечення стійкості підприємства інженер налаштував надлишкові фізичні канали між комутаторами. Протокол STP (Spanning Tree Protocol) автоматично заблокував петлі, і мережа працює стабільно. Однак під час аудиту безпеки виявляється, що весь внутрішній трафік підприємства раптово почав проходити через невідомий комутатор у гостьовій кімнаті, що призвело до критичної деградації швидкості та перехоплення пакетів (MitM).

Суперечність (Проблематизація): між алгоритмічною довірою протоколів резервування, які автоматично перебудовують топологію для підвищення стійкості, та відсутністю вбудованої автентифікації цих алгоритмів, що дозволяє зловмиснику маніпулювати логікою мережі.

Завдання здобувачу:

1. Помислити як зловмисник (моделювання загрози): розробити сценарій генерації фальшивих BPDU-фреймів з нульовим пріоритетом (Bridge ID = 0).
2. Проаналізувати каскадну деградацію легальної топології під впливом цієї атаки.
3. Спроекувати контрзаходи на рівні доступу (налаштування BPDU Guard та Root Guard) для ізоляції штатного механізму резервування від ворожого втручання.

Методологічний акцент. Руйнування міфу про те, що «внутрішня локальна мережа є безпечним середовищем» та формування навички захисту самих протоколів забезпечення надійності.

Завдання 2. Викрадення шлюзу «Вразливість першого стрибка (FHRP Hijacking)»

Тема заняття: Резервування мереж.

Проблемна ситуація. Мережа налаштована на абсолютну відмовостійкість: два потужні маршрутизатори працюють у парі (протокол HSRP або VRRP), утворюючи єдиний віртуальний шлюз замовчування. Раптом весь вихідний трафік усього офісу зупиняється. Аналіз показує, що зловмисник з підключеного ноутбука почав розсилати HSRP-пакети (Hello packets) з максимальним пріоритетом 255.

Суперечність (Проблематизація): штатний сценарій «перехоплення ініціативи» резервним маршрутизатором у разі відмови головного перетворюється на інструмент відмови в обслуговуванні (DoS) або маршрутизації в «чорну діру».

Завдання здобувачу:

1. Дослідити анатомію протоколу FHRP та виявити його ключову вразливість: трансляцію службових повідомлень у відкритому вигляді (Multicast).
2. Проблематизувати життєздатність системи за умов відсутності перевірки джерела керуючих команд.
3. Впровадити криптографічний захист (MD5 автентифікацію) протоколу резервування та обґрунтувати це рішення з позиції інженерної стійкості.

Методологічний акцент. Трансформація мислення від простого «забезпечення працездатності» (ring ide) до глибокого усвідомлення потенційних векторів компрометації резервних контурів.

Завдання 3. Фрагментація як зброя: «Аномалія деградації ресурсів (Teardrop/Rose)»

Тема заняття: Інтернет протокол.

Проблемна ситуація: сервер компанії успішно витримує легальне навантаження. Проте зловмисник розпочинає специфічну DDoS-атаку на мережевому рівні: він надсилає IP-пакети з навмисно пошкодженими параметрами зсуву фрагментації (Fragmentation Offset), які перекривають один одного. Процесор (CPU) брандмауера та сервера досягає 100% завантаження, намагаючись зібрати ці пошкоджені фрагменти, що призводить до повного зависання заліза (Kernel Panic).

Суперечність (Проблематизація): між легальним, стандартизованим механізмом протоколу IPv4 (фрагментація великих пакетів для передачі вузькими каналами) та його перетворенням на інструмент фатального виснаження апаратних ресурсів приймаючої сторони.

Завдання здобувачу:

1. Розібрати структуру заголовка IPv4 і змоделювати логіку зловмисника при створенні накладених (overlapping) фрагментів.
2. Проаналізувати процес фізичної деградації оперативної пам'яті сервера (буфера збирання пакетів) під час такої атаки.
3. Спроекувати політику безпеки (скидання фрагментованих пакетів або обмеження швидкості їх обробки) для нівелювання цієї алгоритмічної вразливості.

Методологічний акцент. Формування навичок аналізу деградації системи в екстремальних умовах (під впливом аномального трафіку) і проблематизація недоліків базових Інтернет-стандартів.

Завдання 4. Отруєння маршрутів: «Ін'єкція фантомних мереж»

Тема заняття: Інтернет протокол.

Проблемна ситуація. У великій корпоративній мережі працює протокол динамічної маршрутизації (наприклад, OSPF). Внаслідок компрометації одного малозначущого комутатора доступу на складі, зловмисник інтегрує програмний маршрутизатор-шпигун і починає анонсувати фейковий маршрут до критичної підмережі (наприклад, до серверів баз даних) з найкращою метрикою. Уся корпоративна мережа автоматично перенаправляє конфіденційний трафік на склад.

Суперечність (Проблематизація): механізм динамічної самоорганізації мережі, покликаний забезпечити її гнучкість та стійкість, стає ідеальним транспортним середовищем для поширення шкідливих топологічних змін.

Завдання здобувачу:

1. Змоделювати вектор атаки типу «Route Injection» та пояснити алгоритмічну сліпоту легальних маршрутизаторів, які охоче приймають фейкові оновлення.

2. Проблематизувати розрив між логічною топологією мережі та вимогами інформаційної безпеки.

3. Розробити багаторівневий захист: перевести некритичні інтерфейси в пасивний режим (Passive Interface) та налаштувати фільтрацію префіксів (Route Filtering/Prefix Lists).

Методологічний акцент. Розвиток системного бачення, де інженер аналізує наслідки безконтрольного поширення керуючої інформації всередині «довіреного» контуру.

Пакет проблемних завдань

«Контекстуальні та людино-машинні інтерфейси комп'ютерних мереж»

Для розгортання цього пакета завдань використано обов'язковий компонент ОК «Комп'ютерні мережі» Управління мережами – це класичний приклад середовища, де найменша операторська помилка в інтерфейсі управління (CLI) або нерозуміння потреб бізнесу призводить до глобальних аварій.

Завдання 1. Когнітивне перевантаження оператора. «Синтаксична пастка CLI»

Тема заняття: Базові налаштування пристрою. Принципи комутації

Проблемна ситуація: мережевому інженеру-початківцю (оператору) доручено додати новий VLAN (номер 50) на магістральний порт комутатора, де вже працюють 20 інших VLAN-ів критичної інфраструктури. Оператор вводить стандартну команду `switchport trunk allowed vlan 50`. Комутатор мовчки приймає команду без жодних попереджень. Наступної секунди зв'язок з усіма 20 попередніми мережами зникає, підприємство зупиняється (оскільки оператор забув додати слово `add`, і система перезаписала весь список єдиним значенням).

Суперечність (Проблематизація): між бездоганим виконанням машиною синтаксично правильної команди та катастрофічним UX (User Experience) командного рядка (CLI), який не має механізмів захисту від фатальних когнітивних помилок людини в умовах стресу.

Завдання здобувачу:

1. Проаналізувати інцидент не як «помилку недосвідченого працівника», а як фундаментальний недолік людино-машинного інтерфейсу.

2. Розробити методологічний регламент (чек-лист або скрипт автоматизації) для внесення змін у критичну інфраструктуру.

3. Запропонувати способи компенсації високого когнітивного навантаження: перехід від ручного CLI до декларативного управління (Network Automation/Ansible), де інженер описує кінцевий стан, а не кроки.

Завдання 2. Перевизначення вимог: «Ілюзія абсолютної безпеки»

Тема заняття. Маршрутизація. Мережна безпека

Проблемна ситуація. Керівник бізнесу (замовник) вимагає «зробити мережу абсолютно безпечною» і наказує запровадити прив'язку MAC-адрес на кожному порту (Port Security), закрити всі некритичні порти та впровадити складну 802.1X авторизацію. Інженер виконує вимогу. Через тиждень робота офісу паралізована: співробітники не

можуть підключити нові ноутбуки для презентацій у переговорних кімнатах, IT-відділ перевантажений заявками, а генеральний директор не може підключити свій новий планшет і вимагає звільнити адміністратора.

Суперечність (Проблематизація): між технічно досконалою реалізацією безпекових політик та реальним бізнес-контекстом компанії, який вимагає високої гнучкості та мобільності працівників.

Завдання здобувачу:

1. Проблематизувати початковий запит замовника: навчитися виявляти суперечність між бажанням контролю та потребою в зручності.
2. Розробити стратегію ведення переговорів зі стейкхолдером (як пояснити ціну безпеки мовою бізнес-ризиків та UX).
3. Запропонувати компромісне технологічне рішення: сегментація мережі на жорстко контрольовану (Corporate) та гнучку ізольовану (Guest/BYOD) з автоматизованим порталом самореєстрації (Captive Portal).

Завдання 3. Хаос логічної топології: «Анатомія організаційного конфлікту»

Тема заняття Віртуальні локальні мережі

Проблемна ситуація: замовник дає технічне завдання на проектування VLAN: «Бухгалтерія не повинна бачити юристів, але головний бухгалтер має доступ усюди; принтери повинні бути спільними, але тільки для керівників відділів, а системний адміністратор має бачити всіх, крім директора». Спроба студента реалізувати це за допомогою простих VLAN і статичної маршрутизації перетворюється на хаотичне нашарування сотень правил доступу, які неможливо підтримувати.

Суперечність (Проблематизація): спроба інженера прямолінійно перенести хаотичну, неформальну соціальну ієрархію компанії на сувору логічну архітектуру комп'ютерної мережі.

Завдання здобувачу:

1. Зупинити процес налаштування і проблематизувати саме технічне завдання (виявити логічні суперечності).
2. Здійснити інжиніринг вимог: структурувати ролі доступу, замінивши індивідуальні забаганки матрицею рольового управління доступом (RBAC).
3. Захистити нову, спрощену логічну топологію перед «замовником» (викладачем), довівши, що вона виконує 95% критичних бізнес-завдань, але знижує вартість підтримки в 10 разів.

Завдання 4. Конфлікт зворотного зв'язку: «Тіньові метрики стійкості»

Тема заняття Протоколи STP, EtherChannel.

Проблемна ситуація: інженер об'єднав два гігабітні канали в один логічний (EtherChannel) для сервера баз даних. Система моніторингу (UX панелі керування) радісно відображає зелений статус та пропускну здатність «2 Гбіт/с». Керівництво задоволене інвестицією. Однак у години пік сервіс продовжує «гальмувати». Фізичний аналіз показує, що 99% трафіку йде лише по одному кабелю, тоді як інший простоює через те, що алгоритм балансування навантаження за замовчуванням (за MAC-адресою відправника) не підходить для ситуації, коли всі запити йдуть від одного магістрального маршрутизатора.

Суперечність (Проблематизація): між інтерфейсом інструментів моніторингу, які створюють у людини ілюзію вирішеної проблеми (відображаючи теоретичну ємність), та суворою апаратною дійсністю (фактичним алгоритмом хешування).

Завдання здобувачу:

1. Пояснити феномен сліпої довіри оператора до візуальних дашбордів.
2. Проблематизувати алгоритм балансування навантаження в EtherChannel у контексті конкретної бізнес-задачі (трафік від багатьох до багатьох чи від одного до багатьох).
3. Провести розслідування та змінити алгоритм балансування на L3/L4 (за IP-адресою або портами TCP/UDP).

ХАКАТОН «ЗВОРОТНЕ ПРОЄКТУВАННЯ»

Мета: формування методологічних знань та умінь здобувачів через розв'язання інженерної проблеми в умовах невизначеності, відсутності вихідної документації та обмеженого доступу до налаштувань.

Позааудиторний захід

Цільова аудиторія: студенти 3-го курсу спеціальності «Комп'ютерна інженерія».

Тривалість: 4 години.

Хід хакатону

Здобувачам не надається готове технічне завдання з чіткими кроками. Замість цього вони отримують «проблемну ситуацію».

Сценарій: «Ви – команда незалежних мережевих аудиторів. Компанія N звільнила попереднього системного адміністратора через підозру у саботажі. Жодної документації щодо топології мережі не залишилося. Відомо лише, що корпоративна мережа складається з 3 філій, об'єднаних через маршрутизатори, а в центральному офісі періодично виникає "широкомовний шторм" (Broadcast storm) та втрата пакетів до сервера баз даних. Паролі доступу (enable secret) на більшості комутаторів змінені (заблоковані).

Артефакти, що надаються командам:

1. Файл-модель мережі (наприклад, .pkt для Packet Tracer), де пристрої заблоковані пароллями (або доступ є лише до кількох периферійних ПК).

2. Файл-дамп трафіку (PCAP), знятий з одного з портів комутатора в момент аварії.

Основне завдання. Не маючи прямого доступу до конфігурацій (CLI), за допомогою методів мережевої розвідки (Network Reconnaissance) та аналізу трафіку відтворити логічну топологію, знайти причину збою та запропонувати оптимізовану архітектуру.

За наявності технічної можливості можливо дати студентам фізичне обладнання.

Етап 1. Інжиніринг вимог та висунення гіпотез (15 %)

Діяльність команд. Аналіз брифу. Команди проводять мозковий штурм, формують гіпотези щодо можливих причин збою (наприклад, петля маршрутизації, некоректно налаштований STP, конфлікт IP-адрес, зловмисний VLAN hopping).

Результат етапу. Складений план аудиту мережі. Жодних змін у конфігурації не проводиться.

Роль викладача: спостерігач. Викладач лише перевіряє логічність гіпотез і дає дозвіл на перехід до наступного етапу.

Етап 2. Зворотне проєктування та збір метрик (50%)

Діяльність команд. Здобувачі використовують інструменти кінцевих вузлів (ПК): ping, tracer/traceroute, arp -a, nslookup, або інших утиліт, а також аналізують наданий дамп Wireshark. Вони аналізують мережу:

Визначають шлюзи за замовчуванням та маски підмереж.

Відстежують шляхи проходження пакетів для відмальовування маршрутизаторів.

Шукають у дампі трафіку пакети протоколів CDP/LLDP, DTP, VTP, OSPF/EIGRP hello-пакети, щоб зрозуміти, яке обладнання стоїть за межами їхнього доступу.

Результат етапу. Повністю відтворена на папері або в редакторі (draw.io/Visio) логічна схема невідомої мережі (L2 та L3 рівні). Локалізація вузла, де виникає проблема.

Етап 3. Синтез рішення (Відновлення та Оптимізація) (25 %)

Діяльність команд. Після успішного захисту відтвореної топології, викладач "видає" командам паролі від обладнання (або дозволяє зробити процедуру Password Recovery). Команди повинні:

Усунути знайдену проблему (наприклад, налаштувати STP, прибрати зайві лінки, змінити метрики маршрутизації).

Оптимізувати топологію (впровадити агрегацію каналів, сегментувати VLAN, налаштувати ACL для безпеки).

Результат етапу. Працююча, стабільна та безпечна модель мережі, яка відповідає кращим інженерним практикам.

Етап 4. Рефлексія та захист результатів (10 %)

Діяльність команд. Кожна команда презентує результати свого дослідження. Фокус презентації. Не просто показати, що "зараз все працює", а пояснити свій методологічний шлях: як саме вони здогадалися про структуру мережі; які хибні гіпотези були відкинуті; чому нове архітектурне рішення є оптимальним з точки зору ресурсів та безпеки.

Критерії оцінювання

Для оцінювання використовується не бінарна система (працює/не працює), а шкала вимірювання методологічної компетентності:

Аналітичний компонент (30 балів). Здатність витягувати інформацію з L2/L3 заголовків пакетів (PCAP). Точність відтворення логічної топології "наосліп". Вміння користуватися діагностичними утилітами ОС.

Проектно-синтетичний компонент (30 балів). Ефективність усунення суперечності (збою). Обґрунтованість вибору нових протоколів (наприклад, чому OSPF замість RIP). Дотримання принципів відмовостійкості.

Дослідницько-рефлексивний компонент (20 балів). Логіка вибудови гіпотез на першому етапі. Здатність команди пояснити причинно-наслідкові зв'язки між конфігурацією та апаратною поведінкою мережі.

Командна взаємодія (20 балів). Ефективний розподіл ролей (наприклад: "аналітик трафіку", "архітектор топології", "інженер з безпеки").

Високий рівень 90-100; середній 70-89; низький менше ніж 70.

Поради організатору

Уникайте прямих підказок. Якщо команда зайшла в глухий кут, ставте навідні запитання: "Зверніть увагу на TTL у цих пакетах, що він вам каже про кількість транзитних вузлів?" або "Чи бачите ви в дампі STP BPDU пакети? Хто є root bridge?".

Ефект несподіванки. На третьому етапі можна ввести додаткову змінну (ввідну): наприклад, повідомити, що компанія щойно придбала нову філію, і її потрібно терміново інтегрувати в розроблену архітектуру без простою основних сервісів. Це перевірить гнучкість інженерного мислення.

ІНСТРУКЦІЯ ДЛЯ СТУДЕНТА

Вітаємо на хакатоні! Цей захід відрізняється від класичних лабораторних робіт. У вас не буде покрокової інструкції, готової топології чи повного доступу до налаштувань. Ваше завдання – діяти в умовах невизначеності, використовуючи своє інженерне мислення та методологічний підхід для розв'язання реальної проблеми.

Ваша місія (Легенда). «Ви – команда незалежних мережевих аудиторів (DFIR-фахівців та мережевих архітекторів). Компанія «N» звернулася до вас із критичною проблемою: у їхній корпоративній мережі періодично виникає «широкомовний шторм» (Broadcast storm), що призводить до втрати пакетів і відмови в обслуговуванні сервера баз даних.

Ускладнення: попередній системний адміністратор звільнився, не залишивши жодної документації. Паролі enable secret на ключових маршрутизаторах та комутаторах змінені. Доступ до фізичного обладнання відсутній (воно знаходиться у віддалених філіях або заблокованих серверних).

Що у вас є: файл віртуального середовища (Cisco Packet Tracer / GNS3) це ваш «чорний ящик». Ви маєте доступ лише до консолей кількох периферійних ПК працівників; файл dump.pcap дампу трафіку, знятий з одного з портів комутатора в момент аварії, який можна проаналізувати у Wireshark.

Етап 1. Аналіз проблеми та висунення гіпотез

Не поспішайте вводити команди. Уважно вивчіть симптоми. Проведіть мозковий штурм у команді. Результат: короткий список із 2-3 технічних гіпотез та план дій для їх перевірки.

Етап 2. Зворотне проєктування або «Розвідка наосліп»

Використовуйте доступні утиліти на кінцевих ПК (ping, tracert, arp -a, ipconfig /all) та аналізуйте дампи трафіку у Wireshark. Шукайте службові пакети: OSPF Hello, STP BPDU, CDP/LLDP. Заборонено: намагатися зламати паролі на цьому етапі або скидати налаштування обладнання. Результат: відтворена на папері (або в draw.io) детальна логічна схема мережі. Ви маєте точно вказати, де саме знаходиться вузьке місце або помилка конфігурації. Після успішного захисту схеми перед експертом (викладачем), ви отримаєте паролі доступу.

Етап 3. Синтез рішення та оптимізація

Отримавши доступ до CLI пристроїв, усуньте знайдену проблему. Але це не все. Ваше завдання – оптимізувати мережу. Результат: працездатна, стабільна віртуальна мережа.

Етап 4. Підготовка до захисту (Пітчінг)

Підготувати 5-хвилинну презентацію вашого рішення. Важливо! Замовника цікавить не стільки те, яку команду ви ввели, скільки як ви дійшли до цього рішення. Розкажіть про ваш методологічний шлях.

3. Правила виживання та взаємодії

Розподіл ролей. Одразу розподіліть зони відповідальності. Наприклад: Аналітик трафіку (працює з Wireshark та логами). Мережевий інженер (тестує з'єднання, будує топологію). Капітан/Архітектор (координує дії, перевіряє гіпотези на логічність, готує презентацію).

Запити до ментора. Викладач/ментор не дає прямих відповідей. Ви можете ставити запитання лише у форматі, що передбачає відповідь «Так/Ні», або просити підтвердити логіку вашого дослідження.

Фіксуйте все: ведіть «журнал інциденту», записуйте, які IP-адреси виявили, які хибні гіпотези відкинули. Це стане основою вашого фінального захисту.

Критерії оцінювання вашої команди

Експертне журі оцінюватиме вас не за швидкість, а за якість інженерного мислення:

Глибина деконструкції: наскільки точно ви змогли відмалювати архітектуру, спираючись лише на непрямі дані (мережеву розвідку).

Ефективність синтезу: Чи дійсно запропонована вами нова конфігурація усуває проблему і є безпечною?

Методологічний контекст: Як ви працювали з суперечностями? Чи змогли ви пояснити розрив між теоретичною роботою протоколу і його реальною поведінкою у цьому сценарії?

Командна взаємодія та захист: професіоналізм пітчінгу, аргументація вибору інструментів та злагодженість дій.

Бажаємо успіху!

Можливі інші теми для хакатонів

Для ОК «Системне програмування». Фокус зворотного проєктування: Декомпіляція та аналіз низькорівневого коду (асемблер) або поведінковий аналіз системних процесів.

Інструментарій: кумулятори процесорів, зневаджувачі (debuggers), дизасемблери.

Сценарій: командам надається скомпільований бінарний файл (чорний ящик), який виконує певну роботу з пам'яттю (наприклад, специфічний алгоритм обробки даних). Потрібно дослідити його виконання в оперативній пам'яті, відновити логіку роботи та написати власний, оптимізований аналог (C/C++ або Python), що працюватиме швидше.

Для ОК «Технології проєктування цифрових систем». Фокус зворотного проєктування: відновлення логіки апаратного забезпечення за часовими діаграмами.

Інструментарій: VHDL-симулятори (ModelSim), Proteus або Logisim.

Сценарій: надається часова діаграма сигналів (timing diagram) на входах і виходах невідомого цифрового пристрою. Команди мають розробити граф-схему автомата, синтезувати його на VHDL та довести через симуляцію, що їхній пристрій генерує ідентичні сигнали.

МЕТОД ПОРТФОЛІО

Мета: формування навичок рефлексії, здатності до самоаналізу та критичної оцінки власних досягнень через розв'язання інженерної проблеми в умовах невизначеності, відсутності вихідної документації та обмеженого доступу до налаштувань.

Позааудиторна робота.

Цільова аудиторія: студенти спеціальності «Комп'ютерна інженерія».

Тривалість: протягом вивчення ОК.

Сучасна інженерна діяльність вимагає від фахівця з комп'ютерної інженерії не лише володіння інструментами розробки чи налаштування апаратної інфраструктури, а й здатності системно мислити, виявляти приховані технічні суперечності та знаходити оптимальні архітектурні рішення. Інженерно-дослідницьке портфоліо є персональним цифровим простором здобувача, де фіксується не просто фінальний результат роботи, а безпосередньо сам процес дослідження, логіка вибору технологій та алгоритми подолання апаратних і програмних збоїв.

Портфоліо має наскрізну модульну структуру. Це означає, що воно наповнюється вами поетапно в межах вивчення окремих освітніх компонентів (ОК), де за кожен сформований інженерний кейс здобувач отримує поточну оцінку. На завершальних етапах навчання розрізнені модулі інтегруються в єдину, цілісну екосистему, яка демонструє вашу здатність, як майбутнього інженера вирішувати комплексні міждисциплінарні завдання.

Структура інженерного кейсу

Замість традиційного додавання звичайних звітів із лабораторних чи практичних робіт, кожен технічний кейс у портфоліо має бути розгорнутий як мінідослідження і містити такі обов'язкові складники:

- **Обґрунтування проблеми (Проблематизація).** Опис вихідних умов, обмежень та виявленої технічної суперечності. Наприклад, ситуація, коли теоретична ємність проєктуємих каналів зв'язку суперечить реальним показникам пропускну здатності апаратної частини.

- **Формулювання гіпотези та вибір технологічного стеку.** Теоретичне обґрунтування того, чому саме обрана топологія, протокол маршрутизації (наприклад, OSPF замість RIP) чи парадигма програмування дозволять розв'язати проблему ефективно.

- **Експериментальна верифікація та моделювання.** Покроковий опис процесу симуляції чи створення віртуальних полігонів (середовища типу Cisco Packet Tracer, емулятори мікропроцесорів, Docker-контейнери тощо). У цей розділ обов'язково додаються графічні схеми, архітектурні діаграми та витяги з системних логів.

- **Трасування помилок та відлагодження (Debugging).** Критично важливий розділ. Здобувач повинен задокументувати лог виникнення багів чи апаратних збоїв (траблшутинг), детально проаналізувати причини, чому початкова конфігурація або код не спрацювали, та описати методологію усунення цих відхилень.

- **Аналіз результатів та екстраполяція.** Зіставлення фінальних метрик із вихідною гіпотезою, оцінювання меж масштабованості системи та формулювання аргументованого рішення щодо доцільності розгортання розробленої моделі на реальній фізичній інфраструктурі.

Технічна реалізація. Платформи та інструменти

Здобувач має право обрати один із двох інструментальних варіантів розгортання портфоліо, узгодивши його з викладачем на початку вивчення дисципліни.

Варіант А. Платформа контролю версій GitHub / GitLab

Цей формат є пріоритетним, оскільки він занурює процес навчання у реальний виробничий контекст ІТ-індустрії.

1. Створюється головний публічний репозиторій з назвою за шаблоном: engineering-portfolio-[Прізвище].

2. У корені репозиторію створюється головний файл README.md, який містить загальну інформацію про розробника, мапу портфолію та гіперпосилання на папки окремих дисциплін.

3. Для кожного окремого освітнього компонента (наприклад, ОК «Комп'ютерні мережі», ОК «Архітектура комп'ютерів», ОК «Об'єктно-орієнтоване програмування») виділяється ізольована директорія першого рівня.

4. Опис інженерних кейсів усередині папок виконується за допомогою мови розмітки Markdown. Файли кодів конфігурацій, скрипти асемблера чи класи на Python мають супроводжуватися розлогими коментарями та оформлюватися у pre-форматовані блоки коду.

Варіант Б. Хмарне середовище Google Docs

Цей формат орієнтований на глибоке текстове та табличне документування інженерних рішень.

1. На Google Диску створюється коренева папка з налаштованим доступом за посиланням (у режимі перегляду/коментування для викладачів).

2. Кожен освітній компонент ведеться в межах окремого текстового документа. Обов'язковим є використання ієрархічних стилів заголовків (Заголовок 1, Заголовок 2) для автоматичного збирання інтерактивного змісту.

3. Порівняльні таблиці параметрів, метрики навантажувального тестування та логі конфігурацій мають структуруватися за допомогою вбудованих HTML-таблиць із чітким вирівнюванням та виділенням заголовків.

Міждисциплінарна інтеграція. Ключова цінність технології портфолію полягає у переході від ізольованого вивчення предметів до формування цілісного інженерного бачення. Отримавши бали за портфолію в межах поточного ОК, ви не закриваєте цей матеріал, а використовуєте його як базис для наступних розробок. Наприклад, під час вивчення ОК «Об'єктно-орієнтоване програмування» ви розробляєте модульне програмне забезпечення. Наступного семестру, у межах ОК «Комп'ютерні мережі», ви розгортаєте мережеву інфраструктуру, куди інтегруєте цей раніше написаний код, зв'язуючи модулі взаємними гіперпосиланнями. У головному файлі портфолію (чи навігаційному документі Google Docs) має з'явитися наскрізний розділ «Міждисциплінарні проекти», де ви аналізуєте, як низькорівнева логіка (схемотехніка, асемблер) впливає на високорівневі процеси (паралельні обчислення, хмарні сервіси чи інформаційну безпеку).

Критерії та рубрики оцінювання

Критерій оцінювання портфолію	Високий рівень	Середній рівень	Низький рівень
Когнітивний компонент			
Глибина аналізу проблеми та гіпотез	Технічна суперечність чітко виділена, надано розгалужене теоретичне обґрунтування вибору конкретної архітектури, алгоритму чи протоколу з детальним порівнянням альтернатив.	Проблема визначена, але аналіз неповний. Обґрунтування рішення спирається на базові теоретичні відомості; альтернативи згадані побіжно, без глибокого порівняння.	Проблема описана поверхнево за принципом «дано завдання – виконано завдання». Альтернативні варіанти рішень не розглядалися, теоретичне обґрунтування відсутнє.

Рефлексивний компонент			
Логіка відлагодження (Troubleshooting) та самоаналіз	Детально зафіксовано процес виникнення помилок (багів, колізій), наведено логи відмов та системно, з критичним осмисленням власного досвіду, описано послідовність дій для приведення системи до стабільного стану.	Помилки частково зафіксовані, проте їх аналіз має фрагментарний характер. Описано кроки виправлення, але бракує усвідомлення першопричин збоїв (виправлення здебільшого методом спроб і помилок).	Усі помилки приховані, описано виключно бездоганний фінальний результат без демонстрації ходу виправлення збоїв та здатності критично оцінювати власні дії.
Праксеологічно-технологічний компонент			
Якість моделювання та інженерної верифікації	Віртуальні полігони побудовані коректно, з дотриманням галузевих стандартів. Надано вичерпний аналіз отриманих експериментальних даних, метрики оформлені в чіткі графіки або структурні таблиці.	Моделювання виконано в цілому правильно, але існують некритичні технологічні огріхи. Дані представлені, проте їх систематизація та візуалізація (графіки, таблиці) потребують доопрацювання.	Експериментальна частина зведена до мінімуму або представлена випадковими скріншотами без пояснення взаємозв'язків між елементами. Модель відірвана від реальних умов.
Комунікативний компонент			
Міждисциплінарна інтеграція та культура документування	Продемонстровано наскрізні логічні зв'язки з іншими ОК. Наявна чітка деревоподібна структура репозиторію. Витримано високу професійну культуру коментування коду, академічну доброчесність та стандарти технічного тексту.	Зв'язки з іншими дисциплінами простежуються слабо. Структура файлів зрозуміла, але коментування коду чи оформлення тексту містить стилістичні та організаційні недоліки.	Директорії хаотичні, файли мають несистемні назви. Відсутні будь-які логічні зв'язки між суміжними інженерними дисциплінами. Професійна лексика ігнорується.
Мотиваційний компонент			
Проактивність та дослідницька ініціатива	Здобувач демонструє стійкий інтерес до вирішення проблеми, виходить за межі базового завдання, самостійно випробовує нетипові інженерні рішення та ініціативно досліджує додаткові технології.	Здобувач виконує завдання в повному обсязі, виявляє базовий інтерес до роботи, проте діє суворо за методичними вказівками, рідко проявляючи власну креативність чи ініціативу.	Робота виконується формально, виключно для отримання мінімального балу. Зацікавленість у розширенні знань, пошуку нових підходів або оптимізації рішень повністю відсутня.

Регулярне та якісне ведення портфоліо дозволить вам сформувати об'єктивну карту власних здобутків, яка стане основою вашою конкурентною перевагою під час взаємодії з майбутніми роботодавцями.

Бажаємо успіху!

ФРАГМЕНТ ПОРТФОЛІО ЗДОБУВАЧА ОСВІТИ

Здобувач вищої освіти: Іван Д.

Спеціальність: 123 «Комп'ютерна інженерія»

ОК: «Комп'ютерні мережі»

Платформа розгортання полігону: Cisco Packet Tracer

КЕЙС 1. Комплексне забезпечення безпеки каналного рівня (Layer 2 Access Security)

1. Проблематизація та технічна суперечність. У базових конфігураціях комутаторів сімейства Cisco Catalyst усі порти за замовчуванням перебувають у стані відкритого доступу та належать до VLAN 1. Це створює критичну вразливість перед атаками типу MAC-flooding (переповнення таблиці комутації засобами утиліт на кшталт masof), за яких комутатор починає транслювати вхідні кадри на всі порти (флуд), фактично перетворюючись на хаб. Існує суперечність між необхідністю забезпечення фізичного підключення кінцевих вузлів та ризиком несанкціонованого перехоплення трафіку або підключення стороннього обладнання до незахищених розеток у робочих кабінетах.

2. Гіпотеза та вибір інструментів захисту. Для нейтралізації загроз Layer 2 буде застосовано технологію Port Security з динамічним вивченням адрес (MAC Sticky) та жорсткою політикою реагування на порушення (Violation Shutdown). Усі невикористовувані фізичні інтерфейси мають бути примусово вимкнені та переведені в ізольований сегмент – VLAN 999 (BlackHole), щоб унеможливити запуск атак через вільні порти.

3. Реалізація та фрагмент конфігураційного файлу (Switch CLI)

Налаштування безпеки на робочому комутаторі доступу SW-Access-01 для пулу інтерфейсів користувачів (Fa0/1 - Fa0/10):

```
SW-Access-01# configure terminal
SW-Access-01(config)# vlan 999
SW-Access-01(config-vlan)# name BlackHole_VLAN
SW-Access-01(config-vlan)# exit
```

! Конфігурування активних портів доступу

```
SW-Access-01(config)# interface range fastEthernet 0/1 - 10
SW-Access-01(config-if-range)# switchport mode access
SW-Access-01(config-if-range)# switchport access vlan 10
SW-Access-01(config-if-range)# switchport port-security
SW-Access-01(config-if-range)# switchport port-security maximum 1
SW-Access-01(config-if-range)# switchport port-security mac-address sticky
SW-Access-01(config-if-range)# switchport port-security violation shutdown
SW-Access-01(config-if-range)# exit
```

! Ізоляція та консервація вільних інтерфейсів

```
SW-Access-01(config)# interface range fastEthernet 0/11 - 24, gigabitEthernet 0/1 - 2
SW-Access-01(config-if-range)# switchport mode access
SW-Access-01(config-if-range)# switchport access vlan 999
SW-Access-01(config-if-range)# shutdown
SW-Access-01(config-if-range)# end
SW-Access-01# copy running-config startup-config
```

4. Трасування помилок та відлагодження (Troubleshooting)

Опис інциденту: Під час тестування системи безпеки до порту Fa0/1 замість штатного ПК-01 було підключено тестовий ноутбук із підробленою MAC-адресою. Інтерфейс миттєво перейшов у стан Down, заблокувавши передачу даних.

Аналіз системного логу комутатора:

%PORT_SECURITY-2-PSECURE_VIOLATION: Security violation occurred, caused by MAC address 0001.C9A4.BC72 on port FastEthernet0/1.

%LINEPROTO-5-UPDOWN: Line protocol on Interface FastEthernet0/1, changed state to down

%LINK-5-CHANGED: Interface FastEthernet0/1, changed state to administratively down

Алгоритм відновлення лінійного протоколу після інциденту (виправлення стану err-disabled):

```
SW-Access-01# show interface fastEthernet 0/1 status
```

Port	Name	Status	Vlan	Duplex	Speed	Type
Fa0/1	LAN_Users	err-disabled	10	auto	auto	10/100BaseTX

```
SW-Access-01# configure terminal
```

```
SW-Access-01(config)# interface fastEthernet 0/1
```

```
SW-Access-01(config-if)# shutdown
```

```
SW-Access-01(config-if)# no shutdown
```

```
SW-Access-01(config-if)# end
```

5. Аналіз результатів. Застосування Port Security повністю ліквідувало загрозу неконтрольованого розростання таблиці MAC-адрес (CAM-таблиці). Механізм sticky дозволив автоматично імпортувати діючі MAC-адреси легітимних пристроїв у конфігурацію, позбавивши адміністратора від необхідності ручного введення десятків ідентифікаторів.

КЕЙС 2. Розгортання та оптимізація динамічної маршрутизації OSPFv2 (Single-Area OSPF)

1. Проблематизація та технічна суперечність. Статична маршрутизація в розподілених мережах не має масштабованості та здатності до швидкої конвергенції при виходу з ладу магістральних каналів зв'язку. Проте запуск динамічного протоколу OSPF без належного налаштування інтерфейсів створює дві проблеми: надлишкове навантаження на кінцеві робочі станції через розсилку мультикаст-пакетів 224.0.0.5 (Hello) та загрозу безпеці (можливість перехоплення оновлень таблиць маршрутизації або несанкціонованого підняття OSPF-сусідства зловмисником).

2. Гіпотеза. Використання однообласного OSPF (Area 0) дозволить побудувати відмовостійку логічну топологію мережі. Для блокування генерації службового трафіку в бік користувацьких підмереж без порушення процесів маршрутизації необхідно застосувати директиву passive-interface.

3. Реалізація та фрагмент конфігураційного файлу (Router CLI)

Налаштування центрального маршрутизатора ядра R-Core-01. Мережа розбита на підмережі класу C за допомогою безкласової маски (VLSM).

```
R-Core-01# configure terminal
```

```
R-Core-01(config)# router ospf 10
```

```
R-Core-01(config-router)# router-id 1.1.1.1
```

! Оптимізація трафіку: переведення всіх інтерфейсів за замовчуванням у пасивний стан

```
R-Core-01(config-router)# passive-interface default
```

! Активація надсилання Hello-пакетів виключно на магістральні інтерфейси зв'язку з іншими роутерами

```
R-Core-01(config-router)# no passive-interface GigabitEthernet 0/0/0
```

```
R-Core-01(config-router)# no passive-interface GigabitEthernet 0/0/1
```

! Оголошення мереж із використанням інверсних масок (Wildcard Masks)

```
R-Core-01(config-router)# network 10.0.0.0 0.0.0.3 area 0
```

```
R-Core-01(config-router)# network 10.0.0.4 0.0.0.3 area 0
R-Core-01(config-router)# network 192.168.10.0 0.0.0.255 area 0
R-Core-01(config-router)# network 192.168.20.0 0.0.0.255 area 0
R-Core-01(config-router)# end
```

4. Трасування помилок та відлагодження (Troubleshooting)

Опис інциденту: після внесення змін магістральний інтерфейс Gi0/0/0 не зміг встановити сусідство (Adjacency) з суміжним маршрутизатором R-Branch-01. Стан OSPF залишався в позиції DOWN або INIT.

Команди діагностики та аналіз виводу:

```
R-Core-01# show ip ospf neighbor
! Вивід порожній або сусід застряг у стані INIT
```

```
R-Core-01# show ip ospf interface gigabitEthernet 0/0/0
GigabitEthernet 0/0/0 is up, line protocol is up
Internet Address 10.0.0.1/30, Area 0
Process ID 10, Router ID 1.1.1.1, Network Type BROADCAST, Cost: 1
Timer intervals configured, Hello 10, Dead 40, Wait 40, Retransmit 5
```

Виявлена помилка: На суміжному роутері R-Branch-01 адміністратор помилково змінив таймери за замовчуванням (Hello був виставлений на 15). Оскільки для встановлення OSPF-сусідства значення параметрів Hello та Dead інтервалів мають збігатися на обох кінцях лінку, виник збій.

Рішення проблеми (синхронізація таймерів):

```
R-Branch-01(config)# interface gigabitEthernet 0/0/0
R-Branch-01(config-if)# ip ospf hello-interval 10
R-Branch-01(config-if)# ip ospf dead-interval 40
```

Лог успішної конвергенції мережі:

```
%OSPF-5-ADJCHG: Process 10, Nbr 2.2.2.2 on GigabitEthernet0/0/0 from LOADING to FULL, Loading Done
```

5. Аналіз результатів. Коректне оголошення мереж через маски Wildcard убезпечило роутер від випадкового включення зайвих підмереж в OSPF-процес. Використання `passive-interface default` гарантує, що жоден байт службової інформації протоколу маршрутизації не потрапить у користувацькі сегменти, економлячи пропускну здатність каналів.

КЕЙС 3. Реалізація між-VLAN-маршрутизації (Router-on-a-Stick) та розмежування прав доступу засобами Extended ACL

1. Проблематизація та технічна суперечність

Логічна ізоляція мережевих пристроїв на рівні VLAN успішно обмежує ширококомовний трафік. Проте при налаштуванні між-VLAN-маршрутизації на базі sub-інтерфейсів роутера (Router-on-a-Stick) виникає нова загроза: за замовчуванням маршрутизатор забезпечує повну зв'язність між усіма мережами. Це дозволяє пристроям із гостьового сегмента (VLAN 30) отримувати безперешкодний доступ до критичної інфраструктури – серверів бази даних та керування (VLAN 10). Виникає суперечність між необхідністю надання гостям доступу до мережі Інтернет та обов'язком захистити внутрішні ресурси організації.

2. Гіпотеза. Розгортання розширених іменованих списків контролю доступу (Extended Named ACL) та їх інсталяція на підінтерфейсах маршрутизатора дозволить виконувати глибоку фільтрацію пакетів на основі IP-адрес відправника/одержувача та номерів портів призначення, повністю ізолюючи гостьову зону.

3. Реалізація та фрагмент конфігураційного файлу (Router CLI).

Налаштування між-VLAN-маршрутизації та правил фільтрації на маршрутизаторі R-Core-01:

R-Core-01# configure terminal

! Налаштування інкапсуляції sub-інтерфейсів під різні VLAN

```
interface GigabitEthernet 0/0/2.10
description VLAN_Admin_and_Servers
encapsulation dot1Q 10
ip address 192.168.10.1 255.255.255.0
exit
```

```
interface GigabitEthernet 0/0/2.30
description VLAN_Guest_Network
encapsulation dot1Q 30
ip address 192.168.30.1 255.255.255.0
exit
```

! Створення розширеного списку контролю доступу

```
ip access-list extended RESTRICT_GUEST_ACL
! 1. Дозволити гостям отримувати IP по DHCP (запити до локального серверу на порт 67)
permit udp 192.168.30.0 0.0.0.255 host 192.168.10.10 eq bootps
! 2. Заборонити гостям будь-який IP-доступ до адміністративного сегменту 192.168.10.0/24
deny ip 192.168.30.0 0.0.0.255 192.168.10.0 0.0.0.255
! 3. Дозволити гостям весь інший трафік (наприклад, вихід в Інтернет через NAT)
permit ip 192.168.30.0 0.0.0.255 any
exit
```

! Прив'язка ACL до гостьового інтерфейсу на ВХІД (Inbound)

```
interface GigabitEthernet 0/0/2.30
ip access-group RESTRICT_GUEST_ACL in
end
```

4. Трасування помилок та відлагодження (Troubleshooting)

Опис інциденту: після активації ACL хости з гостьової VLAN 30 втратили можливість відкривати вебсайти за доменними іменами в мережі Інтернет, хоча прямі IP-пінги до зовнішніх DNS-серверів (наприклад, 8.8.8.8) проходили успішно.

Аналіз покрокового виконання правил ACL за допомогою лічильників матчів:

```
R-Core-01# show access-lists RESTRICT_GUEST_ACL
```

```
Extended IP access list RESTRICT_GUEST_ACL
```

```
10 permit udp 192.168.30.0 0.0.0.255 host 192.168.10.10 eq bootps (24 matches)
20 deny ip 192.168.30.0 0.0.0.255 192.168.10.0 0.0.0.255 (112 matches)
30 permit ip 192.168.30.0 0.0.0.255 any (450 matches)
```

Виявлена помилка: локальний DNS-сервер мережі розташований за адресою 192.168.10.10 (у межах VLAN 10). Правило №20 (deny ip) повністю блокувало будь-які UDP-запити на 53-й порт DNS-сервера від гостьових пристроїв, оскільки воно стоїть вище у списку, ніж загальний дозвіл.

Рішення проблеми за допомогою ін'єкції правила з конкретним Sequence Number:

```
R-Core-01# configure terminal
```

```
R-Core-01(config)# ip access-list extended RESTRICT_GUEST_ACL
```

! Додаємо виняток для DNS-запитів на 15-ту позицію перед загальною заборонаю

```
R-Core-01(config-ext-nacl)# 15 permit udp 192.168.30.0 0.0.0.255 host 192.168.10.10 eq domain
```

```
R-Core-01(config-ext-nacl)# end
```

5. Аналіз результатів. Завдяки перевірці логіки роботи списку та використання інструменту послідовних номерів (Sequence Numbers) вдалося тонко налаштувати безпеку периметра. Гостьова мережа повністю ізольована від корпоративних даних (спрацьовує deny), але зберігає критично необхідний доступ до внутрішніх інфраструктурних служб (DHCP, DNS) та зовнішніх веб-ресурсів.

Приклади індивідуальних та групових проєктів

«ДОСЛІДЖЕННЯ, ОПТИМІЗАЦІЯ ТА МОДЕЛЮВАННЯ ВІДМОВСТІЙКОЇ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ В УМОВАХ МАСШТАБУВАННЯ БІЗНЕС-ПРОЦЕСІВ ПІДПРИЄМСТВА»

Формат виконання – індивідуальний дослідницький проєкт

Формулювання проблеми (невизначеність). Студент отримує базовий опис підприємства (наприклад, «Компанія розширює штат і відкриває новий філіал; з'явилися скарги на затримки у передачі даних; керівництво вимагає ізолювати гостьовий Wi-Fi та фінансовий відділ; бюджет обмежений»). Студенту не дається готовий покроковий алгоритм чи готова топологія. Він має самостійно декомпонувати проблему, визначити технічні суперечності, обрати необхідні мережеві протоколи й архітектурні рішення та обґрунтувати свій вибір.

Структура завдання

Проєкт побудовано за логікою науково-інженерного дослідження:

Етап 1. Методологічний аналіз та проєктування адресації. Задача – аналіз вхідних даних та декомпозиція вимог підприємства.

Завдання: самостійний розрахунок оптимального адресного простору за технологією розрахунок VLSM (IPv4) та планування ієрархічної адресації (IPv6). Студент повинен закласти коефіцієнт майбутнього масштабування мережі (на 20-30% хостів) та аргументувати обрані межі підмереж.

Етап 2. Архітектурне моделювання Layer 2 / Layer 3. Задача – створення відмовостійкої топології в Cisco Packet Tracer за умов обмеження фізичних ліній зв'язку та загрози виникнення петель маршрутизації.

Завдання: побудова дворівневої моделі мережі. Налаштування віртуальних мереж (VLAN), безпеки комутації. Вирішення проблеми надлишковості за допомогою EtherChannel (LACP) та RSTP. Впровадження динамічної маршрутизації (OSPFv2) між центральним офісом та філіалом.

Етап 3. Інтеграція мережевих сервісів та кібербезпека. Задача – забезпечення автономності клієнтських пристроїв та захисту критичних даних (серверної зони).

Завдання: розгортання DHCP, DNS, NAT/PAT. Розробка та впровадження списків контролю доступу (ACL). Студент має самостійно визначити точки фільтрації трафіку (інтерфейси та напрямки) для мінімізації навантаження на процесори маршрутизаторів.

Етап 4. Експериментальна верифікація та рефлексія. Задача – симуляція аварійних ситуацій (відмова одного з каналів EtherChannel або вихід з ладу комутатора ядра).

Завдання: проведення стрес-тестування мережі (ping, traceroute, simulation mode). Аналіз поведінки мережі під час збою.

Структура техніко-методологічного звіту

- Техніко-економічне обґрунтування (ТЕО): Чому було обрано саме таку топологію? Які протоколи маршрутизації розглядалися і чому OSPF/RSTP є оптимальними для цієї задачі?
- Математичний розрахунок: таблиці VLSM з обґрунтуванням резерву адрес.
- Графічна модель: схема логічної та фізичної топології.
- Специфікація налаштувань: ключові лістинги команд CLI з текстовими коментарями автора (пояснення логіки команди, а не просто копіпаст).
- Протокол тестування та відлагодження: опис виявлених під час моделювання помилок (траблшутинг) та способів їх усунення.
- Висновки та рефлексія. Самооцінка досягнення мети, визначення обмежень створеної моделі.

Критерії оцінювання

Компонент оцінювання	Критерії формування компетентності	Макс. бали
1. Методологічний підхід та аналіз (Аналітика)	Глибина аналізу проблеми, коректність і системність декомпозиції вимог, точність математичного розрахунку VLSM без надлишковості адресації.	20
2. Інженерна якість моделі (L2/L3)	Працездатність топології, правильність конфігурування VLAN, забезпечення відмовостійкості (STP, EtherChannel), коректність сходження протоколу OSPF.	35
3. Рішення з безпеки та сервісів (Захист і NAT)	Логіка побудови ACL (мінімізація дірок безпеки), коректність роботи NAT/PAT та базових інфраструктурних сервісів (DHCP, DNS).	15
4. Якість дослідницького звіту (Культура тексту)	Структурованість, наявність інженерних обґрунтувань (ТЕО), грамотність коментування лістингів, опис процесу пошуку помилок (траблшутингу).	10
5. Захист проекту (Проблемна співбесіда)	Демонстрація здатності аргументувати власні технічні рішення. Швидка реакція на зміну умов викладачем (наприклад: «А що зміниться в ACL, якщо ми додамо ще один підрозділ?»).	20
РАЗОМ		100

«ПРОЄКТУВАННЯ, КОМПЛЕКСНА ІНТЕГРАЦІЯ ТА АВТОМАТИЗАЦІЯ РОЗПОДІЛЕНОЇ ВІДМОВСТІЙКОЇ МЕРЕЖЕВОЇ ІНФРАСТРУКТУРИ МУЛЬТИФІЛІАЛЬНОГО ПІДПРИЄМСТВА»

Формат виконання – груповий дослідницький проєкт

Склад команди – 3–4 здобувачі вищої освіти.

Примітка: певні елементи наприклад хмари умовно імітуються.

Розподіл ролей та сфер відповідальності в команді

Для виключення ситуацій, коли проєкт робить одна особа, кожен учасник обирає та захищає одну з чотирьох функціональних ролей. Кожна роль має чітко визначений інженерний ділянок та персональні зони відповідальності

Роль 1. Головний мережевий архітектор (Lead Network Architect / Team Lead). Зона відповідальності – архітектура Layer 2 / Layer 3, адресація, відмовостійкість backbone-каналу.

Завдання: Розробка єдиного планового адресного простору (VLSM) для всієї розподіленої мережі з урахуванням перспективи розширення. Побудова та конфігурування комутації ядра та доступу (VLAN, Trunk, EtherChannel, RSTP). Налаштування міжфіліальної динамічної маршрутизації (OSPF / EIGRP) та резервних маршрутів.

Методологічний фокус – оптимізація топології, запобігання колізіям адресації між підмережами командних партнерів.

Роль 2. Інженер з мережевої безпеки та доступу (Network Security & Access Specialist). Зона відповідальності: політики безпеки, периметр мережі, контрольований доступ, бездротовий сегмент.

Завдання: Розробка та імплементація межмережеских екранів та розширених списків доступу (ACL) для ізоляції критичних зон (бухгалтерія, сервери, гостьовий Wi-Fi). Налаштування трансляції адрес (Dynamic NAT/PAT, Static NAT для зовнішніх серверів). Організація захищених бездротових мереж (Enterprise Wi-Fi, WPA2/WPA3) та L2-захист (Port Security, DHCP Snooping, Dynamic ARP Inspection).

Методологічний фокус: баланс між безпекою інфраструктури та її функціональною доступністю для користувачів.

Роль 3. Інженер мережевих сервісів та автоматизації (Network Services & DevOps Engineer). Зона відповідальності: Інфраструктурні сервіси, моніторинг, скрипти адміністрування.

Завдання: Централізоване розгортання та конфігурування сервісів DHCP (DHCP Relay/Helper-address), DNS, Web/FTP, NTP. Налаштування систем централізованого збору логів (Syslog) та моніторингу стану обладнання (SNMP). Автоматизація базових завдань конфігурування або перевірки стану мережі (використання Python-скриптів або внутрішніх інструментів автоматизації).

Методологічний фокус: неперервність надання сервісів, мінімізація ручного конфігурування при масштабуванні.

Роль 4. Інженер із якості, відлагодження та документації (QA & Troubleshooting Lead). (У команді з 3 осіб ця функція розподіляється між ролями 1–3). Зона відповідальності: тестування відмовостійкості, верифікація рішень, підготовка підсумкової технічної документації.

Завдання: Розробка програми та методики випробувань (Test Plan) для всієї мережі. Моделювання та проведення стрес-тестів: штучний розрив ліній (Failover test), атаки на виснаження DHCP-пулу, переповнення MAC-таблиць. Локалізація та усунення колізій, що виникають під час об'єднання модулів, розроблених іншими учасниками.

Методологічний фокус – критична оцінка працездатності комплексної системи, системний траблшутинг.

Проблемний сценарій для групового виконання (кейс)

Ситуаційне завдання. Міжнародна холдингова компанія об'єднує Центральний офіс, Віддалений виробничий філіал та Хмарну інфраструктуру

Суперечності та обмеження, які має розв'язати команда. Конфлікт вимог до трафіку: виробничий філіал генерує безперервний трафік датчиків та відеоспостереження, який не повинен перевантажувати канал зв'язку з Центральним офісом.

Аварійна стійкість. При обриві основного оптоволоконного каналу між офісами мережа має автоматично переключатися на резервний без втрати сесій критичних сервісів.

Гнучкість доступу. Зовнішні підрядники мають отримувати тимчасовий доступ виключно до тестового сервера у хмарі, не бачачи внутрішньої мережі підприємства.

Масштабованість. Схема адресації та маршрутизації має дозволити підключення ще 2 філіалів без перебудови існуючих налаштувань.

Етапи взаємодії та рефлексії в команді

Етап 1. Узгодження архітектурного контракту. Команда проводить спільний аналіз проблеми, фіксує таблицю розподілу IP-простору та специфікацію інтерфейсів (міжавтономні межі). Результат – підписаний усіма учасниками «Контракт адресації та топології».

Етап 2. Автономна розробка модулів. Кожен учасник налаштовує свій сегмент (архітектор – маршрутизацію, безпечник – ACL/NAT, DevOps – сервіси).

Етап 3. Інтеграційне збирання та розв'язання конфліктів. Об'єднання конфігурацій в єдину модель. На цьому етапі неминує виникають помилки взаємодії (непроходження OSPF, заблокований ACL DHCP-трафік тощо), які команда має вирішити колегіально.

Етап 4. Стрес-тестування та захист. Проведення випробувань та командний захист.

Критерії оцінювання групового проєкту

Оцінка є комбінованою та складається з командного та індивідуального блоків:

Компонент оцінювання	Критерії та показники методологічної компетентності	Макс. бали
1. Системність та цілісність інфраструктури (Командна оцінка)	Працездатність всієї мережі як єдиної системи: узгодженість OSPF/VLAN між сегментами, працездатність резервних каналів, коректність роботи сервісів на наскрізних маршрутах.	30
2. Індивідуальний внесок за обраною роллю (Індивідуальна оцінка)	Глибина та якість виконання персонального завдання (Архітектор / Безпечник / DevOps / QA), відсутність помилок у зонах персональної відповідальності.	30

3. Повнота технічної документації та роадмапу (Командна оцінка)	Наявність спільного технічного звіту, який містить «Контракт адресації», загальну схему, розділи з конфігурацією кожного модуля та протокол тестування.	15
4. Захист проєкту та Cross-Troubleshooting (Індивідуально-командна)	Командний захист. Кожен учасник повинен відповідати не лише за свій модуль, а й розуміти зв'язок із модулями колег. Демонстрація реакції на штучно викликаний викладачем збій у мережі.	25
РАЗОМ		100

«РОЗРОБКА ТА ОПТИМІЗАЦІЯ МУЛЬТИКРОКОВОЇ ПРОМПТ-СИСТЕМИ (AI-АСИСТЕНТА) ДЛЯ ВИРІШЕННЯ ПРИКЛАДНОЇ ФАХОВОЇ ЗАДАЧІ»

Формат виконання – індивідуальний дослідницький проєкт

Проблемна ситуація (контекст). Сучасні генеративні моделі штучного інтелекту (LLM) під час стандартних одноразових запитів (zero-shot) часто дають поверхневі, галюциновані або некоректно структуровані відповіді. Перед здобувачем постає інженерне завдання: виступаючи в ролі промпт-інженера, самостійно обрати актуальну практичну проблему у своїй фаховій або повсякденній діяльності та створити промпт-систему (конверс запитів / пайплайн), яка гарантує стабільний, відтворюваний та якісний результат виконання задачі генеративним ШІ.

Структура та етапи виконання завдання. Проєкт реалізується за п'ятьма послідовними етапами:

Етап 1. Формулювання задачі та вибір цільової моделі. Визначення прикладної проблеми (наприклад: «Автоматизація генерації специфікацій обладнання», «Створення рекурсивного асистента для трешшутингу коду», «Генерація індивідуальних навчальних сценаріїв» тощо). Обґрунтування вибору генеративної моделі (ChatGPT / Claude / Gemini) та її специфічних параметрів (температура, контекстне вікно, системні інструкції).

Етап 2. Проєктування архітектури промпту. Створення комплексної структури системного промпту із застосуванням сучасних технік: Role & Context (призначення чіткої ролі, обмежень та цільової аудиторії); Few-Shot Prompting (надання 2–3 прикладів «вхід-вихід» для калібрування формату); Chain-of-Thought (CoT) (примусове побуквене/крокове міркування моделі для складних обчислювальних або логічних задач); Structured Output Criteria (жорстка фіксація формату виводу: JSON, Markdown-таблиці, YAML, код).

Етап 3. Ітеративна оптимізація та трешшутинг запитів (Prompt Refinement). Проведення серії експериментів (не менше 4–5 ітерацій). Фіксація первинної відповіді ШІ та виявлення недоліків (помилки в логіці, галюцинації, недотримання формату). Внесення виправлень у промпт (додавання негативних обмежень Negative Constraints, технік Role-play, Delimiters тощо). Порівняльний аналіз результатів «до» та «після» корекції.

Етап 4. Оцінка безпеки та стійкості (Safety & Robustness). Перевірка промпт-системи на вразливості. Проведення стрес-тестування на Prompt Injection / Jailbreak (чи можна збити асистента з інструкцій некоректним вхідним даним). Впровадження захисних механізмів (Guardrails) у текстовий промпт для запобігання виходу за межі визначеної теми.

Етап 5. Оформлення підсумкового логбуку та захист. Складання технічного звіту (Prompt Engineering Logbook) та публікація готової промпт-системи.

Структура звіту

Звіт повинен демонструвати дослідницьку рефлексію студента і містити такі розділи:

- Титульний лист (із зазначенням теми, варіанта/задачі, ПІБ здобувача та викладача).
- Мета проєкту та опис обраного кейсу.
- Використана генеративна модель (із зазначенням версії та параметрів).
- Карта промпт-архітектури (System & User Prompts). Повний текстовий код підсумкового промпту (з виділенням блоків: Роль, Контекст, Завдання, Приклади, Обмеження).

- Протокол ітеративного розвитку (Журнал експериментів): Таблиця порівняння ітерацій (Ітерація № -> Зміна в промпті -> Зміна у відповіді ШІ -> Оцінка ефективності).
- Приклади 2–3 успішних та 1 несусвітнього/невдалого результату із поясненням причин.
- Аналіз етичності та безпеки. Опис потенційних ризиків (академічні, етичні, авторські права) та способів їх нейтралізації у проєкті.
- Висновки та рефлексія. Самооцінка досягнення мети та оцінка ефективності взаємодії «людина-ШІ».

Критерії оцінювання проєкту

Компонент оцінювання	Критерії та показники майстерності	Макс. бали
1. Проблемний аналіз та складність задачі	Актуальність обраного кейсу, складність вимог до вихідного контенту, чіткість визначення цілей.	15
2. Інженерна якість та техніка промптингу	Застосування просунутих методик (Chain-of-Thought, Few-Shot, Structured Formatting, System Instructions), відсутність двозначностей.	30
3. Дослідницька ітеративність (Траблшутинг)	Глибина опису процесів відлагодження промпту, наявність виправлених помилок, порівняльний аналіз ітерацій у логбуку.	20
4. Стійкість та захищеність системи (Robustness)	Наявність захисних інструкцій від некоректного вводу, відсутність галюцинацій у вихідних даних.	15
5. Захист проєкту та демонстрація (Live Demo)	Усне обґрунтування прийнятих інженерних рішень, демонстрація роботи промпту в реальному часі, відповіді на запитання викладача.	20
РАЗОМ		100

«КОМАНДНА РОЗРОБКА КОМПЛЕКСНОГО АІ-АСИСТЕНТА (МУЛЬТИКРОКОВОЇ ПРОМПТ-СИСТЕМИ) ДЛЯ РОЗВ'ЯЗАННЯ ПРИКЛАДНИХ ЗАДАЧ ЗА МЕТОДОЛОГІЄЮ AGILE»

Формат виконання: груповий дослідницький проєкт

Склад команди: 3–4 здобувачі вищої освіти.

Створення сучасних АІ-продуктів на основі великих мовних моделей (LLM) вимагає не поодинокого «вгадування» вдалого запиту, а системного інженерного підходу. У реальних ІТ-командах розробка промптів відбувається ітеративно: від формування вимог замовника до проєктування ланцюжків запитів, їх тестування на вразливості та оцінки якості. Проєкт імітує реальний цикл розробки АІ-продукту в команді із застосуванням Agile-практик.

Розподіл AGILE-ролей у команді

Для забезпечення рівномірного залучення та формування фахових компетентностей кожен учасник команди обирає одну з обов'язкових Agile-ролей:

Product Owner (PO) / Фаховий аналітик. Формує візію продукту та описує користувацькі історії (User Stories). Складає та пріоритезує Product Backlog (перелік вимог до АІ-асистента). Приймає готові фічі під час Sprint Review та відповідає за відповідність результату бізнес-потребі.

Scrum Master / Agile Coach. Відповідає за організацію командної роботи, дотримання таймінгу спринтів та проведення стендапів (Daily/Weekly Standups). Веде командну дошку задач (Jira, Trello або Notion). Локалізує та усуває «блокери» (командні чи технічні перешкоди).

Prompt Architect / LLM Engineer (1-2 особи). Проєктує системні промпти, будує ланцюжки міркувань (Chain-of-Thought, Tree-of-Thoughts). Підбирає методи контекстного навчання (Few-Shot, Role-Playing, Delimiters). Проводить ітеративний рефакторинг та оптимізацію тексту запитів.

AI Safety & QA Engineer (Інженер із якості та безпеки). Розробляє метрики оцінки якості відповідей (Evals). Проводить стрес-тестування на Prompt Injection, Jailbreak та галюцинації. Налаштовує обмеження (Guardrails) та контролює формат вихідних даних (JSON, Markdown, YAML).

Адаптація AGILE-процесу (сценарій спринтів)

Проект реалізується через послідовні ітеративні спринти.

Спринт 0. Ініціалізація та налаштування середовища. Sprint Planning: опис ідеї AI-асистента (наприклад: «Асистент для перевірки коду на відповідність стандартам», «Генератор індивідуальних освітніх траєкторій», «Автоматизований аналітик нормативних актів»). Backlog Refinement: створення дошки у Trello/Notion. Формування задач у форматі User Stories: «Як [користувач], я хочу [отримувати відповідь у форматі JSON], щоб [автоматично інтегрувати її у таблицю]».

Спринт 1. Розробка MVP (Мінімально життєздатного продукту) (1 тиждень). Розробка: створення базового системного пром프트 (Zero-Shot) для виконання основної функції. Daily Standups: коротка фіксація (пророблені задачі, плани, блокери). Sprint Review 1 (Демо): внутрішня демонстрація первинних відповідей ШІ. Фіксація недоліків (нестабільність формату, поверхневі відповіді). Sprint Retrospective: обговорення: Що пройшло добре? Що потребує покращення в промпт-архітектурі?

Спринт 2. Оптимізація, Безпека та Рефакторинг (1 тиждень). Інженерне вдосконалення: впровадження Few-Shot прикладів для стабілізації виводу. Додавання Chain-of-Thought (CoT) для складних аналітичних кроків. Впровадження Guardrails (захист від виходу за рамки теми та некоректного вводу). QA & Safety Testing: стрес-тестування симуляційними промптами-атаками. Фіналізація: узгодження підсумкової документації.

Структура командного технічного звіту

Звіт подається в електронному вигляді й повинен містити такі розділи:

- Паспорт Agile-проекту: назва проекту, склад команди із зазначенням Agile-ролей; посилання на інтерактивну Agile-дошку (Trello / Notion / Jira).
- Product Backlog та User Stories: перелік історій користувачів та критерії їх прийняття (Acceptance Criteria).
- Архітектура промпт-системи (System & User Prompts): повний підсумковий текст системного пром프트.
- Опис використаних технік (Role, Few-Shot, CoT, Output Formatting).
- Артефакти Agile-процесу (Журнал спринтів): скріншоти дошки задач на початку та в кінці кожного спринту.
- Короткі звіти з ретроспектив (Retrospective Log).
- Протокол ітеративних змін пром프트 («Версія 1.0 (MVP)» до «Версія 2.0 (Final)»).
- Протокол тестування та безпеки (QA Report): приклади успішних та нейтралізованих некоректних вхідних даних (Prompt Injections).
- Висновки та рефлексія командної взаємодії.

Критерії оцінювання групового проекту

Оцінювання розподіляється між командним результатом та індивідуальним внеском:

Елемент оцінювання	Критерії та показники	Макс. бали
1. Інженерна якість промпт-системи (Командна)	Складність та ефективність архітектури пром프트 (CoT, Few-Shot, Guardrails), відсутність галюцинацій, чітка структурованість виводу.	30
2. Дотримання Agile-методології (Командна)	Якість ведення Agile-дошки, чіткість Product Backlog, наявність зафіксованих спринтів та ретроспектив.	20
3. Надійність та QA-тестування (Командна)	Стійкість системи до некоректного вводу, наявність перевірки на вразливості, наявність метрик оцінки (Evals).	15
4. Індивідуальний внесок за роллю (Індивідуальна)	Якість виконання обов'язків відповідно до обраної ролі (Product Owner / Scrum Master / Prompt Architect / QA Engineer).	20
5. Командний захист та Live Demo (Спільна)	Демонстрація роботи AI-асистента в реальному часі, готовність відповісти на некоректні запити викладача, рефлексія щодо командної роботи.	15
РАЗОМ		100

ДОВІДКИ ПРО ВПРОВАДЖЕННЯ



НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

03041, м. Київ, вул. Героїв Оборони, 16, навчальний корпус № 15, тел./факс: (044) 527-83-52,

E-mail: igor-bolbot@nubip.edu.ua

29.05.2026

№ 10

На № _____ від _____

Довідка

про впровадження результатів дослідження
МАТІЄВСЬКОГО Володимира Валерійовича,

з теми «Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання» на здобуття наукового ступеня доктора філософії за спеціальністю 011 Освітні, педагогічні науки в освітній процес факультету інформаційних технологій Національного університету біоресурсів і природокористування України

Результати дисертаційної роботи МАТІЄВСЬКОГО Володимира Валерійовича «Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання» упроваджувалися в освітній процес факультету під час підготовки майбутніх фахівців з комп'ютерної інженерії на освітньому ступені бакалавр.

Упровадження матеріалів дисертації здійснювалось шляхом упровадження розроблених педагогічних умов формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання. Були запропоновані пакети проблемних завдань, кейсів різного типу (аналітичні, техніко-діагностичні, проектно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), дослідницьких проектів, мотиваційний тренінг «Керуй алгоритмом: мистецтво методологічного мислення в IT», методичне забезпечення міні-хакатонів та ін. Запропоновано комплекс діагностичних методик для визначення рівня сформованості методологічної компетентності майбутніх фахівців з комп'ютерної інженерії.

Використання запропонованих В. В. Матієвським рекомендацій дозволило змістити акцент із репродуктивного засвоєння технічної інформації на самостійний пошук, системний аналіз та генерування нестандартних рішень. Це сприяло підвищенню рівня пізнавальної самостійності студентів та їхньої здатності застосовувати наукові методи при розв'язанні задач, які виникають перед комп'ютерними інженерами.

Апробація результатів дослідження відбулася на базі дисциплін «Комп'ютерні мережі», «Основи промпт-інжинірингу», «Адміністрування комп'ютерних мереж» та під час виконання відповідних курсових робіт. Успішність апробації дозволяє рекомендувати ці матеріали для підготовки майбутніх комп'ютерних інженерів у закладах вищої освіти.

Декана факультету



Виконавець: Болбот І.М.
Тел.: 527-83-52



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ ЗАКЛАД «ЛУГАНСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА»

вул. Віктора Новікова, 2, м. Лубни, 37500, тел. +38 (050) 681 68 05,
e-mail: mail@luguniv.edu.ua, web: luguniv.edu.ua, код ЄДРПОУ 02125131

02.06.2026

№ 1/522

На №

від

АКТ

про впровадження результатів дисертаційної роботи
Матієвського Володимира Валерійовича
«Формування методологічної компетентності майбутніх
фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого
навчання»,
поданої на здобуття ступеня доктора філософії зі спеціальності
011 Освітні, педагогічні науки
в освітній процес ДЗ «Луганський національний університет
імені Тараса Шевченка»

У ДЗ «Луганський національний університет імені Тараса Шевченка» у 2024-2026 рр. впроваджувалися результати дисертаційної роботи В. В. Матієвського «Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання».

Практичне значення мають розроблені автором завдання в рамках навчальних дисциплін «Інформаційні (комп'ютерні) мережі», «ВЕБ технології» та ін. щодо інженерної діяльності, спрямовані на формування навичок структурування проблем та критичного системного аналізу; кейси різного типу (аналітичні, техніко-діагностичні, проєктно-конструкторські, ситуаційно-стратегічні, кейси ретроспективного аналізу), симуляційні та рольові ігри («Система під тиском: випробування в умовах криз», «Технологічний стартап: Від ідеї до реалізації»), методичні матеріали для проведення семінарів-дискусій, дослідницьких проєктів, захисту проєктів у форматах «Технічне демо та взаємоперевірка», «Scrum-спринт», та ін.

Матеріали дисертації використовуються під час проведення лекційних та практичних занять, а також при підготовці курсових робіт студентами спеціальності «Комп'ютерна інженерія».

Результати впровадження підтвердили підвищення рівня пізнавальної самостійності студентів та їхньої здатності застосовувати наукові методи при розв'язанні задач, які виникають перед комп'ютерними інженерами; сприяли зростанню показників успішності з фахових дисциплін тощо.

Зазначене вище підтверджує ефективність результатів дослідження В. В. Матієвського та доцільність їх упровадження в інших закладах вищої освіти України.

Ректор



Олена КАРАМАН

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ДЕРЖАВНИЙ УНІВЕРСИТЕТ
«КИЇВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»
ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК
ТА ТЕХНОЛОГІЙ



MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
STATE UNIVERSITY
«KYIV AVIATION INSTITUTE»
FACULTY OF COMPUTER SCIENCE
AND TECHNOLOGY

03058, Ukraine, Kyiv, pr. Liubomyra Huzara, 1, room 1, aul. 416, 414
tel. +38(044)406-73-96, e-mail: fcst@npp.kai.edu.ua
<https://fcst.nau.edu.ua/>

03058, Ukraine, Kyiv, 1 Liubomyra Huzara Ave, build. 1, room 416, 414
phone +38(044)406-73-96, e-mail: fcst@npp.kai.edu.ua
<https://fcst.nau.edu.ua/>

04.05.2022 № 1.14/15g

Довідка
про впровадження результатів дослідження
МАТІЄВСЬКОГО Володимира Валерійовича,
на тему «Формування методологічної компетентності майбутніх фахівців
з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання»
на здобуття наукового ступеня доктора філософії
за спеціальністю 011 «Освітні, педагогічні науки»
в освітній процес Факультету комп'ютерних наук та технологій
Державного університету «Київський авіаційний інститут»

Результати дисертаційного дослідження МАТІЄВСЬКОГО Володимира Валерійовича на тему: «Формування методологічної компетентності майбутніх фахівців з комп'ютерної інженерії засобами технологій проблемно зорієнтованого навчання» на здобуття ступеня доктора філософії за спеціальністю 011 «Освітні, педагогічні науки» були впроваджені в освітній процес під час підготовки майбутніх фахівців з комп'ютерної інженерії на освітньому ступені бакалавр.

Було впроваджено розроблений мотиваційний тренінг «Керуй алгоритмом: мистецтво методологічного мислення в IT», симуляційний ІІІ-практикум «Інженерне мислення: точка зламу» (діалог із ІІІ-співрозмовником). З метою інтеграції фундаментальних методологічних знань із профільними дисциплінами комп'ютерної інженерії розроблено пакет міждисциплінарних проблемних завдань і кейсів різного типу, зокрема завдань із проблематизації інженерної діяльності. Запропоновані автором методичні матеріали було впроваджено в низці дисциплін професійного циклу. Завдяки методичним рекомендаціям В. В. Матієвського навчання було орієнтовано на самостійну аналітичну роботу та пошук інноваційних рішень. У результаті зросла пізнавальна автономія студентів та їхнє вміння використовувати наукові методи для вирішення фахових інженерних задач.

Результати впровадження підтвердили практичну значущість і ефективність запропонованих у дослідженні методик, тренінгів та діагностичних інструментів і можуть бути рекомендовані для подальшого використання в освітньому процесі закладів вищої освіти при підготовці фахівців за спеціальністю комп'ютерна інженерія.

Декан ФКНТ



Андрій ФЕСЕНКО