

РЕАЛІЗАЦІЯ МІЖПРЕДМЕТНИХ ЗВ'ЯЗКІВ МІЖ ОСВІТНІМИ КОМПОНЕНТАМИ «ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ» І «ЯКІСТЬ ТА ТЕСТУВАННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ» В ПРОЦЕСІ ПІДГОТОВКИ ФАХІВЦІВ З ІНЖЕНЕРІЇ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

Шпортько О. В.

кандидат технічних наук, доцент,
доцент кафедри інформаційних систем та обчислювальних методів
ПВНЗ «Міжнародний економіко-гуманітарний університет
імені академіка Степана Дем'янчука»;
докторант центру аспірантури та докторантури
Національного університету водного господарства та природокористування
ORCID ID: 0000-0002-4013-3057

У статті обґрунтовано доцільність використання елементів міждисциплінарного випереджувального навчання в процесі підготовки бакалаврів спеціальності F2 «Інженерія програмного забезпечення» шляхом формування елементів понятійного апарату наступної дисципліни під час практичного вивчення попередньої дисципліни. Зокрема, наведено методику подання рівнів тестування в дисципліні «Об'єктно-орієнтоване програмування» (ООП). Наголошено на тому, що ця методика не лише формально використовує міждисциплінарні елементи випереджувального навчання, але й сприяє засвоєнню класифікації рівнів тестування і ефективного використанню її в майбутньому. Висвітлено роль структурно-логічної схеми освітньо-професійної програми у формуванні компетентностей та досягненні програмних результатів навчання. Запропоновано можливі застосування методу випереджувального навчання в структурно-логічних схемах освітньо-професійних програм зображувати двонапрямленими стрілками. Продемонстровано практичний підхід до вивчення принципів ООП на прикладі завдання розроблення візуального застосунку-калькулятора та інтеграції його у спільний проєкт конкурсу калькуляторів здобувачів освіти всього потоку. Наведено фрагменти програми для реалізації поставленого завдання мовою програмування C# на платформі NET Framework в оболонці Microsoft Visual Studio Community.

Визначено виконавців кожного рівня тестування для розроблення проєкту конкурсу калькуляторів. Наголошено на тому, що елементи міжпредметного випереджувального навчання інтенсифікують навчальний процес, знижують рівень стресу, готують міцну базу для подальшого розуміння складних понять, дають змогу здобувачам вищої освіти краще розуміти необхідність вивчення наступних дисциплін і підвищують зацікавленість в їх опрацюванні, а встановлення міжпредметних зв'язків з елементами випереджувального навчання забезпечує послідовне формування професійних компетентностей майбутніх фахівців з інженерії програмного забезпечення та підвищує практичну спрямованість освітнього процесу.

Ключові слова: практично-орієнтоване навчання, структурно-логічна схема освітньо-професійної програми, випереджувальне навчання.

Shportko O. V. Implementation of interdisciplinary connections between the educational components “Object-oriented programming” and “Software quality and testing” in the process of training software engineers

The article substantiates the expediency of using elements of interdisciplinary anticipatory training in the process of training bachelors of the F2 “Software Engineering” specialty by forming elements of the conceptual apparatus of the next discipline during the practical study of the previous discipline. In particular, the method of presenting testing levels in the discipline “Object-oriented programming” (OOP) is given. It is emphasized that this methodology not only formally uses interdisciplinary elements of anticipatory learning, but also contributes to the assimilation of the classification of testing levels and its effective use in the future. The role of the structural and logical scheme of the educational and professional program in the formation of competences and the achievement of program learning outcomes is highlighted. It is

proposed that possible applications of the method of anticipatory learning in structural and logical diagrams of educational and professional programs be depicted by two-way arrows. A practical approach to studying OOP principles is demonstrated using the example of the task of developing a visual calculator application and integrating this application into a joint project of a calculator competition for students of the entire stream.

The following are fragments of the program for the implementation of the task in the C# programming language on the .NET Framework platform in the Microsoft Visual Studio Community shell. Performers of each level of testing have been identified for the development of the calculator competition project. It is emphasized that the elements of interdisciplinary anticipatory learning intensify the educational process, reduce the level of stress, prepare a solid base for further understanding of complex concepts, enable students of higher education to better understand the need to study the following disciplines and increase interest in their study, and the establishment of interdisciplinary connections with the elements of anticipatory learning ensures the consistent formation of professional competencies of future software engineering specialists and increases the practical orientation of the educational process.

Keywords: *practically-oriented training; structural and logical scheme of the educational and professional program; advanced training.*

Вступ. В процесі викладання будь-якої дисципліни дуже важливо, щоб її теоретичні положення підкріплювалися практичними прикладами їх застосування. Ще краще, коли ці приклади можуть використовуватися здобувачами освіти у повсякденному житті. Це підвищує рівень засвоєння матеріалу здобувачами освіти та дає змогу в повному обсязі опанувати окремі програмні результати навчання. Проте сьогодні практичної орієнтованості окремих дисциплін для якісної підготовки здобувачів вищої освіти бакалаврського рівня недостатньо. Адже ще й потрібно за весь цикл підготовки сформувати компетентності та досягнути програмних результатів навчання, передбачених освітньо-професійною програмою (ОПП). Задля цього в кожній ОПП чільне місце відводиться структурно-логічній схемі [10]. Саме структурно-логічна схема забезпечує [8]:

- логічну послідовність навчання: визначає, які дисципліни є базовими, а які вивчаються на їх основі;
- реалізацію міжпредметних зв'язків: демонструє, як знання та компетентності з одних освітніх компонентів використовуються в інших;
- формування компетентностей та досягнення програмних результатів навчання: дає змогу простежити, яким чином окремі дисципліни забезпечують досягнення цих показників;
- узгодження змісту ОПП: допомагає уникнути дублювання матеріалу та прогалин у підготовці здобувачів освіти;

– планування освітнього процесу: визначає розподіл дисциплін за семестрами, пріоритетизувати для їх засвоєння.

Сьогодні структурно-логічні схеми ОПП передбачають міжпредметні зв'язки лише між дисциплінами попередніх семестрів з дисциплінами поточного і наступних семестрів.

Моделі міжпредметної інтеграції розробляються в педагогіці з часів Я. Коменського [6], який ще у XVII столітті обґрунтував принципи взаємозв'язку знань. Сучасні моделі інтеграції, розроблені закордонними дослідниками R. Fogarty та J. Stoehr [12], пропонують багаторівневі підходи до об'єднання змісту навчальних дисциплін на основі спільних тем, проблем і проектної діяльності. Питання міжпредметної інтеграції у вищій школі досліджені Л. Лук'яною [3], а підходи до інтеграції фундаментальних, професійних і практичних дисциплін у підготовці фахівців розроблені І. Козловською [2]. Проблеми ж випереджувального навчання наразі розглядаються переважно в межах окремого предмету чи дисципліни. Ми ж пропонуємо розширити міжпредметні зв'язки, ширше впроваджуючи в навчальний процес метод випереджувального навчання [7]. Використовуючи цей метод для підготовки бакалаврів, рекомендуємо формувати елементи понятійного апарату наступної дисципліни під час практичного вивчення попередньої.

Отже, метою статті є пропонування методики використання елементів випереджувального навчання між різними освітніми компонентами в процесі практичної підготовки

у ЗВО. Основні завдання дослідження впливають з поставленої мети:

- навести приклад та обґрунтувати доцільність можливого використання елементів випереджувального навчання між різними освітніми компонентами ОПП підготовки бакалаврів спеціальності F2 «Інженерія програмного забезпечення»;
- запропонувати фрагменти програми мовою програмування C# для демонстрації практичного застосування теоретичного матеріалу під час використання елементів випереджувального навчання;
- зазначити спосіб відображення елементів випереджувального навчання між різними освітніми компонентами ОПП в структурно-логічній схемі.

Методи та методики дослідження.

В процесі опрацювання літературних джерел та проблематики статті використано такі загальні теоретичні педагогічні методи, як аналіз та синтез, узагальнення, порівняння, абстрагування, індукція та дедукція. Для дослідження сучасного стану міжпредметного використання елементів випереджувального навчання застосовувався емпіричний метод педагогічного спостереження.

Результати. Сьогодні дисципліни, пов'язані з вивченням об'єктно-орієнтованого програмування (ООП) та тестування програмного забезпечення (ПЗ), входять у всі відомі нам ОПП підготовки бакалаврів за спеціальністю F2 «Інженерія програмного забезпечення». Причому тестування ПЗ вивчається після ООП, тому що тестувати потрібно вміти насамперед власні розроблені програми. Отже, наведемо приклад практичного використання принципів ООП [9] та випереджувального вивчення при цьому рівнів тестування ПЗ.

Покажемо, наприклад, як основні принципи об'єктно-орієнтованого програмування можна продемонструвати на прикладі застосування-калькулятора. Зрозуміло, що калькулятори використовуються здобувачами освіти і як окремі приладдя, і в складі програмного забезпечення мобільних телефонів, персональних комп'ютерів та інших електронних пристроїв. Принципи їх використання загальновідомі. Отже, створення свого застосунку-

калькулятора з додатковими можливостями (наприклад, з реалізацією алгоритмів [11]) – цікаве практичне завдання. Ми розробляємо нескладні калькулятори Windows-форми на платформі NET Framework в оболонці Microsoft Visual Studio Community мовою програмування C# на одній з лабораторних робіт (наприклад, як на рис. 1), після чого здобувачі освіти публікують їх у загальнодоступному репозиторії на GitHub та мають можливість порівняти свій калькулятор з калькуляторами інших здобувачів потоку (рис. 2). Однак на перших лекціях, коли вивчаються принципи ООП, ми вважаємо доцільним продемонструвати один з калькуляторів здобувачів освіти старших курсів і на його прикладі пояснити застосування цих принципів на практиці.

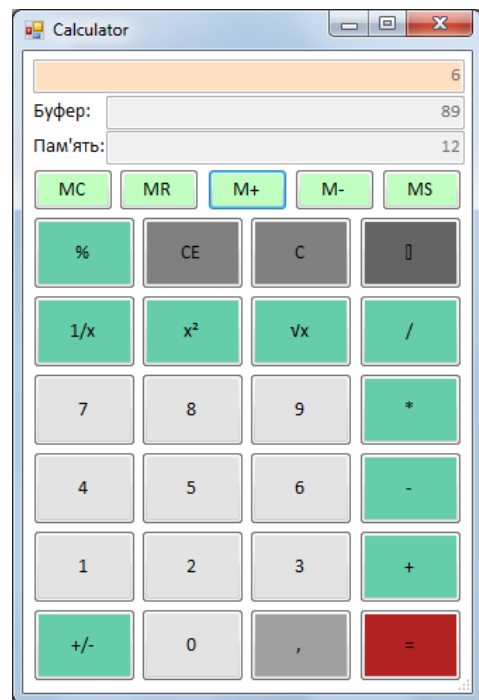


Рис. 1. Калькулятор здобувача освіти Мосейчука Владислава з групи КН-22 ПВНЗ «МЕГУ імені академіка Степана Дем'янука»

Як відомо, ООП базується на чотирьох основних принципах [1].

1) Абстракція. Подаємо її як здатність описувати об'єкти оточуючого світу засобами вибраної мови програмування в рамках поставленого завдання. В процесі програмування виділяється і описується кожен тип об'єктів (клас), а під час завантаження й виконання програми створюються об'єкти вибраного типу. В застосунку-

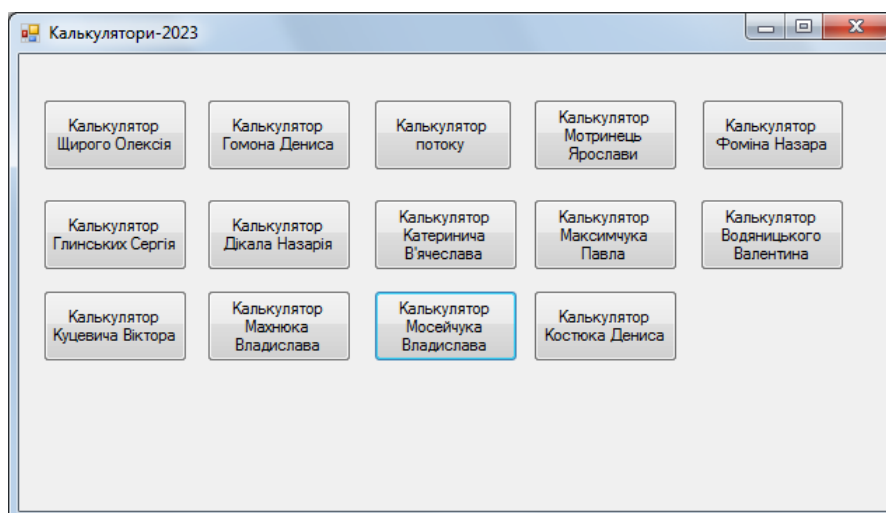


Рис. 2. Головна кнопкова форма проєкту конкурсу калькуляторів між здобувачами освіти потоку

калькуляторі ми описуємо один клас калькулятора, а в процесі експлуатації користувач може відкрити (створити) декілька однотипних об'єктів-калькуляторів.

2) Наслідування. Це здатність одного типу (класу) об'єкта бути породженим від іншого типу (класу). Зрозуміло, що калькулятор ми не розробляємо «з нуля». Навіть завантаження початкового застосунку Windows-форми спричиняє появу порожнього вікна, над яким вже можна виконувати всі базові операції з вікнами (переміщення, зміна розміру, згортання/розгортання тощо). Це автоматично реалізується наслідуванням класу форми-калькулятора від класу *Form*:

```
public partial class FormCalc : Form
{
    bool resultTablo = false;
    string prevOperation = "";

```

```
public FormCalc()
{
    InitializeComponent();
}

```

```
private void plusTablo(string plus)
{
    if (resultTablo)
    {
        textTablo.Text = "0";
        resultTablo = false;
    }
    if (textTablo.Text == "0")
        textTablo.Text = "";
    textTablo.Text += plus;
}

```

```
private void plusTablo()
{
    plusTablo("0");
}
...
}

```

Вже з цього коду здобувач освіти запам'ятовує, де зазначається батьківський клас, а де – клас-нащадок;

3) Інкапсуляція. На наш погляд, вона передбачає:

- здатність поєднання полів, властивостей і методів їх обробки в описі самого класу;
- незалежність об'єкта (всі потрібні поля й методи мають бути в описані в класі об'єкта, а всі зайняті ресурси – звільнені під час його знищення);
- здатність класу приховувати від зовнішнього світу внутрішні деталі реалізації та відображати лише поля, властивості й методи, які можуть бути використані безпосередньо або для подальшого розроблення.

Тут, аналізуючи клас калькулятора, ми акцентуємо увагу на цілісності його коду та на описі окремих методів з модифікаторами *private* чи *public*. Далі пояснюємо, що перші з них можуть викликатися лише зсередини класу, а другі – як зсередини, та і ззовні.

4) Поліморфізм. На перших лекціях ми подаємо його у спрощеному варіанті у рамках спеціалізованого поліморфізму [4] як здатність методів з однаковою назвою, але з різною кількістю чи типами цих аргументів (різною сигнатурою) реалізовувати різну логіку обробки даних. В класі калькулятора цей принцип ми демонструємо на прикладі методу *plusTablo* (див. код вище), який з параметром дописує в табло зазначений текст, а без параметрів – 0.

Вивчаючи тестування ПЗ, найчастіше виділяють такі його рівні.

1) Модульний. Виконується, як правило, розробником для перевірки функціонування окремих фрагментів коду. Під час виконання поставленого завдання з розроблення застосунку-калькулятора тестування на цьому рівні виконує здобувач освіти, відлагоджуючи як окремі можливості, так і свій калькулятор (модуль) загалом.

2) Інтеграційний. Реалізується розробником або тестувальником для перевірки взаємодії фрагментів коду між собою. В нашому завданні цей рівень імплементує теж здобувач освіти, інтегруючи свій калькулятор в репозиторій потоку (рис. 2).

3) Системний. Полягає в перевірці дієздатності системи загалом. В нашому завданні таке тестування виконується кожним здобувачем освіти під час перевірки калькуляторів колег.

4) Приймальний. Найчастіше виконується замовником ПЗ для прийняття рішення про вихід релізу чи повернення до етапу розроблення. В завданні розроблення проєкту курсу калькуляторів на цьому рівні тестує ПЗ викладач.

Бачимо, що, виконуючи завдання з розроблення застосунку-калькулятора, здобувач вищої освіти безпосередньо причетний до трьох з чотирьох рівнів тестування і розуміє

призначення четвертого рівня. Таким чином, подаючи рівні тестування ПЗ в дисципліні ООП, ми не лише формально використовуємо міждисциплінарні елементи випереджувального навчання, але й сприяємо засвоєнню цієї класифікації і ефективному використанню її в майбутньому.

В ОПП міждисциплінарні зв'язки, які відображаються в структурно-логічній схемі, найчастіше вказують на змістову інтеграцію навчальних дисциплін [10]: кожна напрямлена стрілка від дисципліни попереднього семестру до дисципліни наступного семестру свідчить про використання матеріалу попередньої дисципліни та її попередників під час вивчення наступної дисципліни. Наприклад, на рис. 3 бачимо, що знання про об'єктно-орієнтовані моделі використовуються в дисципліні «Якість та тестування ПЗ» за посередництва предмета «Командна розробка та управління ІТ-проєктами». Ми пропонуємо можливі застосування методу випереджувального навчання в структурно-логічних схемах ОПП зображувати двонапрямленими стрілками.

Висновки. За результатами проведеного дослідження можемо дійти таких висновків.

Використання міждисциплінарних елементів випереджувального навчання інтенсифікує навчальний процес, знижує рівень стресу та готує міцну базу для подальшого розуміння

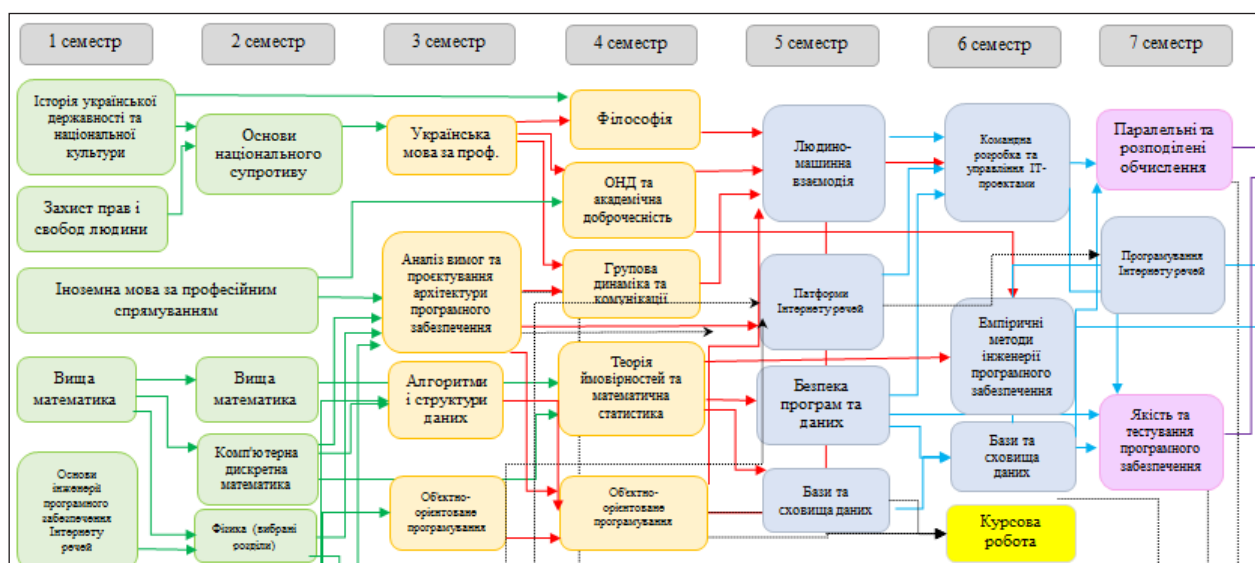


Рис. 3. Фрагмент структурно-логічної схеми ОПП «Інженерія програмного забезпечення Інтернету речей» в ПВНЗ «МГУ імені академіка Степана Дем'янука»

складних понять. Це дає змогу здобувачам вищої освіти краще розуміти необхідність вивчення наступних дисциплін.

Встановлення міжпредметних зв'язків з елементами випереджувального навчання забезпечує послідовне формування професійних компетентностей майбутніх фахівців з інженерії програмного забезпечення та підвищує практичну спрямованість освітнього процесу.

Дисципліна «Об'єктно-орієнтоване програмування» є фундаментальною основою для опанування дисципліни «Якість та тестування програмного забезпечення», оскільки саме об'єктно-орієнтовані моделі визначають структуру програмного продукту, яка надалі

стає об'єктом контролю якості та тестування. З іншого боку, засадничі поняття тестування, такі як його рівні, варто випереджувально подавати в курсі ООП для підвищення зацікавленості у вивченні тестування.

Наочна демонстрація принципів ООП дає змогу здобувачам освіти краще засвоїти теоретичний матеріал, а в подальшому, на лабораторних заняттях, – реалізувати ці принципи на практиці.

Використання штучного інтелекту. Під час підготовки статті інструменти штучного інтелекту використовувалися виключно для інших допоміжних завдань, без використання згенерованих ними об'єктів як результатів академічної діяльності.

Список використаних джерел

1. Григорович В. Об'єктно-орієнтоване програмування. Ч. 1. Львів: Магнолія 2006, 2023. 284 с. ISBN: 978-617-574-244-0.
2. Козловська І. Метапредметна інтеграція як засіб формування змісту професійної освіти. *Інформаційно-телекомунікаційні технології в сучасній освіті: досвід, проблеми, перспективи*: зб. наук. праць. Львів: ЛДУ БЖД, 2009. С. 71–73.
3. Ничкало Н., Лук'янова Л., Хомич Л. Інтердисциплінарна взаємодія у професійній підготовці вчителя як педагогічна реальність. *Міждисциплінарні дослідження складних систем*. 2021. № 19. С. 94–108. DOI: <https://doi.org/10.31392/iscs.2021.19.094>.
4. Поліморфізм (програмування) – Вікіпедія. URL: [https://uk.wikipedia.org/wiki/Поліморфізм_\(програмування\)](https://uk.wikipedia.org/wiki/Поліморфізм_(програмування)) (дата звернення: 6.08.2026).
5. Рівні тестування – Q & A. URL: <https://qlearning.com.ua/theory/lectures/material/testing-levels> (дата звернення: 6.08.2026).
6. Сміт Д. Ян Амос Коменський: завбачливий реформатор шкільної освіти. Київ: MAPXO, 2017. 111 с. ISBN: 978-966-97691-0-7.
7. Швардак М. Особливості технології перспективно-випереджального навчання С.М. Лисенкової. *Інноваційні педагогічні технології у підготовці майбутніх фахівців з вищою освітою: досвід, проблеми, перспективи*: матеріали міжнародної науково-методичної інтернет-конференції, 8–10 жовтня 2013 р. Вінниця, 2013. URL: https://www.researchgate.net/publication/357861063_osoblivosti_tehnologii_perspektivno-viperedzalnogo_navcanna_smlisenkovoї (дата звернення: 6.08.2026).
8. Шпинковський О., Дрозд М. Особливість застосування структурно-логічного уявлення компонентів освітніх програм для формування компетентностей. *Automation Technological and Business Processes*. 2025. Т. 17. № 4. С. 4–10. DOI: <https://doi.org/10.15673/atbp.v17i4.3327>.
9. Шпортко О. Демонстрація основних принципів об'єктно-орієнтованого програмування на прикладі застосування калькулятора. *Цифрові технології в освіті: компетентності та інструменти*: матеріали всеукраїнського науково-педагогічного підвищення кваліфікації, 26 травня – 6 липня 2025 р. Львів; Торунь: Liha-Pres, 2025. С. 137–140. ISBN: 978-966-397-523-8.
10. Щирба В. Структурно-логічна схема підготовки фахівців як конструктивна модель процесу набуття професійних якостей. *Збірник наукових праць Кам'янець-Подільського національного університету імені Івана Огієнка. Серія: Педагогічна*. 2010. Вип. 16. С. 130–132. URL: <https://ped-series.kpnu.edu.ua/article/view/32789/29398> (дата звернення: 6.08.2026).
11. Cormen T.H., Leiserson C.E., Rivest R.L., Stein C. Introduction to Algorithms, Third Edition. Cambridge: MIT Press, 2009. 1320 p.
12. Fogarty R, Stoehr J. Integrating Curricula with Multiple Intelligences: Teams, Themes, and Threads. 2th ed. Thousand Oaks, CA: Corwin, 2007. 232 p. ISBN: 978-1412955539.

References

1. Hryhorovych, V. (2023). Obiektno-orientovane prohramuvannia. Chastyna 1 [Object-Oriented Programming. Part 1]. Lviv: Mahnoliia 2006, 2023. 284 p. ISBN: 978-617-574-244-0 [in Ukrainian].
2. Kozlovskaya, I. (2009). Metapredmetna intehratsiia yak zasib formuvannia zmistu profesiinoi osvity [Meta-subject integration as a means of shaping the content of vocational education]. *Informatsiino-telekomunikatsiini tekhnologii v suchasni osviti: dosvid, problemy, perspektivy: zb. nauk. prats*. Lviv: LDU BZhD, 2009, pp. 71–73 [in Ukrainian].
3. Nychkalo, N., Lukianova, L. and Khomych L. (2021). Interdystsyplinarna vzaiemodiia u profesiinii pidhotovtsi vchytelia yak pedahohichna realnist [Interdisciplinary interaction in teacher training as a pedagogical reality]. *Mizhdystsyplinarni doslidzhennia skladnykh system*. 19, 94–108. <https://doi.org/10.31392/iscs.2021.19.094> [in Ukrainian].
4. Polimorfizm (prohramuvannia) – Vikipediia [Polymorphism (programming) – Wikipedia]. Retrieved from [https://uk.wikipedia.org/wiki/Поліморфізм_\(програмування\)](https://uk.wikipedia.org/wiki/Поліморфізм_(програмування)) (last accessed: 6.08.2026) [in Ukrainian].
5. Rivni testuvannia – Q & A [Testing Levels – Q & A]. Retrieved from: <https://qlearning.com.ua/theory/lectures/material/testing-levels> (last accessed: 6.08.2026) [in Ukrainian].
6. Smit, D. (2017). Yan Amos Komenskyi: zavbachlyvyi reformator shkilnoi osvity [John Amos Comenius: A Visionary Reformer of Schools]. Kyiv: MARKhO, 2017. 111 p. ISBN: 978-966-97691-0-7 [in Ukrainian].
7. Shvardak, M. (2013). Osoblyvosti tekhnologii perspektivno-vyperedzhalnogo navchannia S.M. Lysenkovoi [Features of the technology of prospective and advanced learning by S.M. Lysenkova]. *Innovatsiini pedahohichni tekhnologii u pidhotovtsi maibutnykh fakhivtsiv z vyshchoiu osvitoiu: dosvid, problemy, perspektivy: materialy mizhnarodnoi naukovo-metodychnoi internet-konferentsii, 8–10 zhovtnia 2013 r.* Vinnytsia, 2013. Retrieved from https://www.researchgate.net/publication/357861063_osoblivosti_tehnologii_perspektivno-viperedzhalnogo_navcanna_smlisenkovoi (last accessed: 6.08.2026) [in Ukrainian].
8. Shpynkovskiy, O. and Drozd, M. (2025). Osoblyvist zastosuvannia strukturno-lohichnoho uiavlennia komponentiv osvity dlia formuvannia kompetentnosti [Peculiarities of using a structural-logical representation of the components of educational programs for the formation of competencies]. *Automation Technological and Business Processes*. 17 (4), pp. 4–10. DOI: 10.15673/atbp.v17i4.3327 [in Ukrainian].
9. Shportko, O. (2025). Demonstratsiia osnovnykh pryntsyviv obiektno-orientovanoho prohramuvannia na prykladi zastosunku-kalkuliatora [Demonstration of the basic principles of object-oriented programming using the example of a calculator application]. *Tsyfrovi tekhnologii v osviti: kompetentnosti ta instrumenty: materialy vseukrainskoho naukovo-pedahohichnoho pidvyshchennia kvalifikatsii, 26 travnia – 6 lyupnia 2025 r.* Lviv; Torun: Liha-Pres, pp. 137–140. ISBN: 978-966-397-523-8. [in Ukrainian].
10. Shchyrb, V. (2010) Strukturno-lohichna skhema pidhotovky fakhivtsiv yak konstruktivna model protsesu nabuttia profesiinykh yakosti [Structural and logical scheme of training specialists as a constructive model of the process of acquiring professional qualities]. *Zbirnyk naukovykh prats Kamianets-Podilskoho natsionalnoho universytetu imeni Ivana Ohienka. Seriya pedahohichna*. 16, pp. 130–132. Retrieved from <https://ped-series.kpnu.edu.ua/article/view/32789/29398> (last accessed: 6.08.2026) [in Ukrainian].
11. Cormen, T.H., Leiserson, C.E., Rivest, R.L. and Stein, C. (2009). Introduction to Algorithms, Third Edition. Cambridge: MIT Press. 1320 p. [in English].
12. Fogarty, R and Stoehr, J. (2007) Integrating Curricula with Multiple Intelligences: Teams, Themes, and Threads. 2th ed. Thousand Oaks, CA: Corwin, 2007. 232 p. ISBN: 978-1412955539 [in English].

Дата першого надходження статті до видання: 11.07.2026

Дата прийняття статті до друку після рецензування: 08.08.2026

Дата публікації (оприлюднення) статті: 14.10.2026