

ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ  
«ІРПІНСЬКИЙ ФАХОВИЙ КОЛЕДЖ НАЦІОНАЛЬНОГО УНІВЕРСИТЕТУ  
БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ»

Циклова комісія фундаментальних дисциплін та комп'ютерних технологій

ЗАТВЕРДЖУЮ

Заступник директора  
з навчальної роботи

Вікторія СОВА  
2025 року



РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

«Об'єктно-орієнтоване програмування»

(назва навчальної дисципліни)

галузь знань

**12 Інформаційні технології**

(шифр і назва галузі знань)

освітньо-професійна  
програма

**Комп'ютерна інженерія**

спеціальність

**123 Комп'ютерна інженерія**

відділення

**Інформаційних технологій**

(назва відділення)

2025 рік

Робоча програма «Об'єктно – орієнтоване програмування»  
(назва навчальної дисципліни)

для студентів  
за галуззю знань 12 «Інформаційні технології»

спеціальністю 123 «Комп'ютерна інженерія»

освітньо-професійна  
програма Комп'ютерна інженерія

Розробник(и): Борис ШТЕРН, викладач вищої кваліфікаційної категорії

Робоча програма затверджена на засіданні циклової комісії фундаментальних дисциплін і комп'ютерних технологій  
Протокол від 29 серпня 2025 року № 1

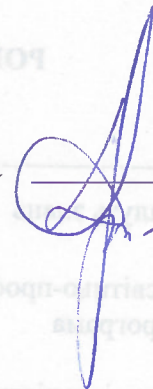
Голова циклової комісії фундаментальних дисциплін і комп'ютерних технологій



Е. Дібрівна

Схвалено методичною радою коледжу.  
Протокол від 29 серпня 2025 року № 1

Голова



Д. Костюк

♥ Ірпінь, 2025 рік

## 1. Опис навчальної дисципліни

Найменування показників	Галузь знань, освітньо-професійна програма, освітньо-професійний ступінь	Характеристика навчальної дисципліни
		денна форма здобуття освіти
Кількість кредитів – 3	Галузь знань: 12 Інформаційні технології	Обов'язкова
Модулів – 2	Спеціальність: 123 Комп'ютерна інженерія Освітньо-професійна програма: Комп'ютерна інженерія	Рік підготовки:
Загальна кількість годин – 90		4-й
		Семестр:
		8-й
Тижневих годин для денної форми здобуття освіти: аудиторних – 4 самостійної роботи – 4	Освітньо-професійний ступінь: фаховий молодший бакалавр	Лекції:
		22 год.
		Практичні, семінарські:
		22 год.
		Лабораторні:
		0 год.
		Самостійна робота:
		46 год.
Вид контролю:		
		Диференційований залік

**Примітка.** Співвідношення кількості годин аудиторних занять до самостійної роботи становить: для денної форми здобуття освіти –44/46

## 2. Мета навчальної дисципліни

Мета дисципліни - отримання студентами знань з області розробки алгоритмів та програмування. Оволодіння такими знаннями дозволить реалізовувати задачі автоматизації обробки інформації, автоматизації керування об'єктами за допомогою комп'ютерної техніки. Такі знання майбутній спеціаліст зможе застосовувати як при подальшому навчанні, так і після отримання вищої освіти у своїй професійній діяльності.

Перелік компетентностей студентів, що формуються в результаті засвоєння дисципліни:

### Загальні компетентності (ЗК):

ЗК 3. Здатність до пошуку, оброблення та аналізу інформації з різних джерел.

ЗК 4. Здатність застосовувати знання у практичних ситуаціях.

ЗК 7. Здатність працювати в команді.

ЗК 8. Здатність вчитися і оволодівати сучасними знаннями.

### Спеціальні компетентності (СК):

СК2. Здатність застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування апаратних, програмних та інструментальних засобів комп'ютерної інженерії.

СК3. Здатність вільно користуватись сучасними комп'ютерними та інформаційними технологіями, прикладними та спеціалізованими комп'ютерно-інтегрованими середовищами для розробки, впровадження та обслуговування апаратних та програмних засобів комп'ютерної інженерії.

СК4. Здатність брати участь у розробці системного та прикладного програмного забезпечення засобів комп'ютерної інженерії з використанням ефективних алгоритмів, сучасних методів і мов програмування.

СК6. Здатність брати участь у модернізації апаратних та програмних засобів комп'ютерної інженерії.

СК10. Здатність аргументувати вибір методів розв'язування спеціалізованих задач, критично оцінювати отримані результати, обґрунтовувати прийняті рішення.

СК12. Здатність створювати, впроваджувати, адмініструвати бази даних і знань з використанням сучасних методів, технологій та систем керування базами даних.

### **3. Передумови вивчення дисципліни**

Дана навчальна дисципліна базується на раніше здобутих результатах навчання таких навчальних дисциплін, як «Комп'ютерна логіка», «Теорія ймовірності та математична статистика», «Дискретна математика», «Програмування».

### **4. Очікувані результати навчання**

#### **Програмні результати навчання (РН):**

РН6. Тестувати, діагностувати та обслуговувати апаратні та програмні засоби комп'ютерної інженерії.РН7. Мати навички розробки, моделювання, тестування, діагностування та обслуговування апаратних та програмних засобів комп'ютерної інженерії.

РН7. Застосовувати знання для формулювання і розв'язування технічних задач спеціальності, використовуючи методи, що є найбільш придатними для досягнення поставлених цілей.

РН9. Розробляти, тестувати, впроваджувати, експлуатувати програмне забезпечення для вбудованих і розподілених систем.

РН10. Здійснювати пошук інформації з різних джерел для розв'язання задач комп'ютерної інженерії.

РН14. Використовувати сучасні інтегровані середовища, методи і технології розробки, впровадження, адміністрування комп'ютерних систем та мереж, баз даних і знань.

РН15. Проводити інсталяцію та налаштування системного та прикладного програмного забезпечення, у тому числі програмних засобів захисту інформації з метою реалізації встановленої політики інформаційної безпеки.

### **5. Критерії оцінювання**

Критерії оцінювання знань студентів наведено в додатку до робочої програми навчальної дисципліни.

### **6. Засоби оцінювання**

Контрольні заходи включають поточний, модульний та підсумковий контроль знань студента.

Поточний контроль здійснюється під час проведення семінарських (практичних) занять та у процесі здійснення самостійної роботи у таких формах: експрес-опитування, тести, задачі, реферати, розрахункові роботи, вирішення ситуаційних завдань, студентські презентації, робота в Інтернет тощо.

Модульний контроль проводиться з метою оцінки результатів навчання студентів на визначених його етапах.

Підсумковий контроль проводиться з метою оцінки результатів навчання на завершальному етапі.

## 7. Програма навчальної дисципліни

### Модуль 1. Основи ООП

#### Тема 1. Основи об'єктно-орієнтованого програмування. Поняття мови програмування Java.

Основні поняття об'єктно-орієнтованого програмування. Принципи ООП. Знайомство з програмним забезпеченням та середовищем програмування. Пакети та бібліотеки середовища програмування. Інтерфейс середовища програмування. Поняття мови програмування. Мова Java. Історія виникнення мови програмування. Основні поняття.

#### Тема 2. Основи класів та об'єктів мови програмування Java.

Поняття класів та об'єктів в мовах програмування. Ключові та зарезервовані слова мови Java. Простий додаток. Бібліотеки.

#### Тема 3. Типи даних та оператори.

Базові типи даних і літерали. Документування коду. Оператори. Оператори керування. Масиви. Поняття циклу. Види циклів. Призначення циклів.

### Модуль 2. Об'єкти мови Java

#### Тема 4. Класи та об'єкти мови Java.

Змінні класів, екземплярів та константи. Обмеження доступу. Конструктори. Методи. Статистичні методи і поля. Модифікатор final. Абстрактні методи. Параметризовані класи та методи. Декомпозиція.

#### Тема 5. Поняття наслідування та поліморфізму. Внутрішні класи. Інтерфейси.

Поняття наслідування. Використання методів final, super, this. Перевизначення методів та поліморфізм. Абстракція та абстрактні класи. Розширення функціональності системи. Клас Object. Внутрішні, вкладені, анонімні класи. Інтерфейси та їх параметризація. Анотації.

#### Тема 6. Поняття «Рядок». Виключення та помилки. Потоки вводу/виводу.

Поняття «рядок». Клас String, StringBuilder, StringBuffer. Регулярні вирази. Форматування рядків. Ієрархія виключень та помилок. Способи обробки виключень. Обробка декількох виключень. Оператор throw та блок finally. Байтові і символічні потоки вводу/виводу. Клас File.

#### Тема 7. Поняття колекцій. Шаблони.

Загальні визначення. Списки. Інтерфейси ListIterator та Comparator. Клас LinkedList та інтерфейс Queue. Інтерфейс Deque та клас ArrayDeque. Алгоритми класу Collections. Поняття шаблону. Шаблони Singleton, Factory method, Prototype, Builder, Abstract Factory, Adapter, Façade.

## 8. Структура навчальної дисципліни

Назви модулів і тем	Кількість годин				
	денна форма				
	усього	у тому числі			
л		п.	інд.	с.р.	
Модуль 1. Основи ООП					
Тема 1. Основи об'єктно-орієнтованого програмування. Поняття мови програмування Java.	11	2	2	-	7
Тема 2. Основи класів та об'єктів мови програмування Java.	13	4	2	-	7
Тема 3. Типи даних та оператори.	19	4	4	-	11
Модульна контрольна робота № 1	2	-	2	-	-
Разом за модулем 1	45	10	10	-	25

Модуль 2. Об'єкти мови Java					
Тема 4. Класи та об'єкти мови Java.	11	4	2	-	5
Тема 5. Поняття наслідування та поліморфізму. Внутрішні класи. Інтерфейси.	9	2	2	-	5
Тема 6. Поняття «Рядок». Виключення та помилки. Потоки вводу/виводу.	13	4	4	-	5
Тема 7. Поняття колекцій. Шаблони.	10	2	2	-	6
Модульна контрольна робота № 2	2	-	2	-	-
Разом за модулем 2	45	12	12	-	21
Усього годин	90	22	22	-	46

### 9. Теми семінарських занять

№ з/п	Назва теми та зміст семінарських занять	Кількість годин
1.	Не передбачено навчальним планом	

### 10. Теми практичних занять

№ з/п	Назва теми і зміст практичних занять	Кількість годин
1	Тема 1. Основи об'єктно-орієнтованого програмування. Поняття мови програмування java. Введення у середовище об'єктно-орієнтованого програмування мови Java	2
2	Тема 2. Основи класів та об'єктів мови програмування Java. Синтаксис мови програмування. Типи даних. Оператори. Введення в класи. Логічні оператори.	2
3	Тема 3. Типи даних та оператори Цикли. Підпрограми та функції. Рекурсія Модульна контрольна робота 1	6
4	Тема 4. Класи та об'єкти мови Java. Класи і об'єкти. Декомпозиція.	2
5	Тема 5. Поняття наслідування та поліморфізму. Внутрішні класи. Інтерфейси. Робота с файлами. Використання принципів ООП - поліморфізм, наслідування.	2
6	Тема 6. Поняття «Рядок». Виключення та помилки. Потоки вводу\виводу. Робота з рядками. Обробка виключень та помилок. Робота з операторами throw/finally.	4
7	Тема 7. Поняття колекцій. Шаблони. Списки. Інтерфейси ListIterator та Comparator. Клас LinkedList та інтерфейс Queue. Інтерфейс Deque та клас ArrayDeque. Алгоритми класу Collections. Поняття шаблону. Шаблони Singleton, Factory method, Prototype, Builder, Abstract Factory, Adapter, Façade. Модульна контрольна робота № 2	4
	Разом	22

### 11. Теми лабораторних занять

№ з/п	Назва теми і зміст лабораторних занять	Кількість годин
1.	Не передбачено навчальним планом	

### 12. Самостійна робота

№ з/п	Назва теми і зміст самостійної роботи	Кількість годин
1	Тема 1. Основи об'єктно-орієнтованого програмування. Поняття мови програмування java. Інструментальні засоби JDK. appletviewer - програма проглядання аплетів Java. jar - програма створення архівів Java. Java - інтерпретатор Java. javac - компілятор Java. javadoc - генератор документації Java.	7
2	Тема 2. Основи класів та об'єктів мови програмування Java. Оператори. Прості типи. Цілі числа. Byte Short Int long. Числа з плаваючою крапкою. Float Double Приведення типу. Автоматичне перетворення типів у виразах. Явне перетворення типів. Символи. Тип boolean.	7
3	Тема 3. Типи даних та оператори Оператори розгалуження та повторення Масиви. Багатовимірні масиви.	11
4	Тема 4. Класи та об'єкти мови Java. Програмування класу, типи класів, структура класу. Змінні представники (instance variables). Оператор new. Оголошення методів. Виклик методу	5
5	Тема 5. Поняття наслідування та поліморфізму. Внутрішні класи. Інтерфейси. Пакети. Трансляція класів в пакетах. Використання пакетів. Інтерфейси. Оператор interface. Оператор implements. Змінні в інтерфейсах. Оператор import. Обмеження доступу.	5
6	Тема 6. Поняття «Рядок». Виключення та помилки. Потоки вводу\виводу. Конструктори. Спеціальний синтаксис для роботи з рядками. Послідовність виконання операторів. Створення рядків. Злиття рядків. Перетворення рядків. Витягання символів. Порівняння. Рівність. Впорядкування.	5
7	Тема 7. Поняття колекцій. Шаблони. Тег HTML <Applet>. Передача параметрів. Контекст аплета - AppletContext і showDocument. Налаштовувальний друк. Порядок ініціалізації аплета. Завдання розмірів графічних зображень. Прості методи класу Graphics. Методи класу Color.	6
	Разом	46

### 13. Індивідуальні завдання

Не передбачено навчальним планом.

### 14. Інструменти, обладнання, програмне забезпечення

При вивченні дисципліни використовується середовище INTELLIJIDEA для створення програм, інтернет сервіси-компілятори Java та навчально-інформаційне середовище MOODLE.

### 15. Розподіл балів, які отримують студенти

7 семестр

Модуль №1 50 балів					Модуль №2 50 балів					Всього	
T1	T2	T3		МКР № 1	T4	T5	T6		T7	МКР № 2	
П.З. № 1	П.З. № 2	П.З. № 3	П.З. № 4		П.З. № 5	П.З. № 6	П.З. № 7	П.З. № 8	П.З. № 9		
8	8	10	9	15	6	8	7	7	7	15	100

### Шкала оцінювання студентів

Рейтинг студента, бали	Оцінка національна за результатами складання екзамену (диференційованого заліку)
90-100	Відмінно
74-89	Добре
60-73	Задовільно
0-59	Незадовільно

### 16. Рекомендовані джерела інформації ОСНОВНА

#### Підручники (навчальні посібники)

1. Вступ до java: теоретичні основи зі світу найпопулярнішої мови програмування. Посібник з навчальної дисципліни об'єктно-орієнтоване програмування для студентів спеціальності 123 «Комп'ютерна інженерія». Голуб. Є.С., Штерн Б.О. -Ірпінь, 2025 - 148с.

2. Перші кроки в Java: навчальний посібник для студентів, які навчаються за спеціальностями "Інформатика", "Програмна інженерія", "Комп'ютерні науки", "Прикладна математика" / О. В. Олецкий - Київ :2017. - 144 с.

3. Підручник Перші кроки програмування мовою Java: навчальний посібник для учнів 10–11 класів (Укр) / Руденко В.Д., Жугастров О.О.- Ранок, 2018 - 112 с.

4. JAVA-програмування. Комп'ютерний практикум. Навчальний посібник для студентів. / Тарнавський Ю.А.- Київ : КПІ ім. Ігоря Сікорського, 2021. – 95 с.

#### ДОПОМІЖНА

5. Head First. Java. Легкий для сприйняття довідник. / Кеті Сьєрра, Берт Бейтс, перекладач Якубовська Г. – Видавництво Фабула, 2022 – 720 с.

#### ІНФОРМАЦІЙНІ РЕСУРСИ В ІНТЕРНЕТІ

6. Освоюємо Java - Вікіпідручник [Електронний ресурс]. – Режим доступу: <https://surl.li/luimib>

7. Java Підручник. Уроки для початківців. W3Schools українською. [Електронний ресурс]. – Режим доступу: <https://w3schoolsua.github.io/java/index.html>

## **КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАНЬ СТУДЕНТІВ З НАВЧАЛЬНОЇ ДИСЦИПЛІНИ «ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ»**

### **Для денної форми здобуття освіти**

Оцінювання знань студентів здійснюється за 100-бальною шкалою (поточний, модульний та підсумковий контроль (диференційований залік).

Робочою програмою навчальної дисципліни передбачено вивчення 2-х модулів обсягом (3 кредити ЄКТС):

1 модуль - 1,5 (45 год.) – 50 балів;

2 модуль - 1,5 (45 год.) – 50 балів.

Робочою програмою навчальної дисципліни передбачено застосування 3-х форм контролю знань студентів: поточного, модульного, підсумкового.

#### **1. Поточний контроль.**

Поточний контроль здійснюється у формі усних відповідей, доповнень на практичних заняттях, письмового опитування, розв'язування задач, виконання тестів тощо.

За кожним елементом модуля, передбаченого робочою програмою, обов'язкова певна форма поточного оцінювання знань.

Такими формами можуть бути:

- письмова контрольна робота (відповіді на питання лекційного курсу, розв'язання задач тощо);
- тестування знань студентів з певного розділу (теми) або з певних окремих питань лекційного курсу;
- виступ на практичних заняттях (з рефератом);
- перевірка розв'язання завдань (задачи, вправи) тощо.

#### **Критеріями оцінки є:**

##### **На практичному занятті оцінюються:**

- вміння аналізувати умову задачі, вміння самостійно шукати шляхи розв'язання;
- правильність побудови алгоритмічних конструкцій, та правильність їх опису на мові програмування;
- реферативні виступи.

##### **При виконанні письмових завдань:**

- повнота розкриття питання;
- цілісність, системність, логічна послідовність, вміння формулювати висновки; акуратність оформлення письмової роботи.

##### **Оцінювання самостійної роботи студента.**

Контроль самостійної роботи студентів здійснюється як під час аудиторних занять (на семінарах, практичних заняттях), так і у позааудиторний час.

Контроль самостійної роботи передбачає:

- визначення ступеня засвоєння матеріалу;
- визначення якості виконання завдань;
- своєчасне виконання і здача поточних завдань;
- оцінку знань, здобутих у результаті самостійної навчальної роботи.

#### **2. Модульний контроль.**

Кожен модуль завершується виконанням студентом модульної контрольної роботи. Модульний контроль є підсумком певного етапу вивчення навчальної дисципліни. Його мета – виявлення проміжних результатів засвоєння студентами змісту навчальної дисципліни. На модульну контрольну роботу передбачено 30% від суми балів, виділених на модуль. Модульна контрольна робота проводиться у тестовій письмовій формі. Критерії оцінювання знань за модульну контрольну роботу наводиться у пояснювальній записці до неї. Оцінка за модуль визначається як сума набраних балів за поточну роботу та за модульну контрольну роботу.

### 3. Підсумковий контроль.

Формою підсумкового контролю з навчальної дисципліни «Об'єктно-орієнтоване програмування» є диференційований залік, який виставляється виключно за результатами поточного та модульного контролю (сума набраних балів за всі модулі). Залік виставляється під час останнього практичного заняття.

Залежно від балів, отриманих за кожний вид навчальної роботи, студент одержує суму балів, яка переводиться в національну оцінку за відповідною шкалою згідно з табл.1:

**Таблиця 1. Переведення рейтингу студента за 100-бальною шкалою в оцінку за національною шкалою**

Рейтинг студента, бали	Оцінка національна
90-100	Відмінно
74-89	Добре
60-73	Задовільно
0-59	Незадовільно

Оцінка «**Відмінно**» виставляється студенту, який систематично працював протягом семестру, показав різнобічні і глибокі знання програмного матеріалу, вмів успішно виконувати завдання, які передбачені програмою, засвоїв зміст основної та додаткової літератури, усвідомив взаємозв'язок окремих розділів навчальної дисципліни, їхнє значення для майбутньої професії, виявив творчі здібності у розумінні та використанні навчально-програмного матеріалу, проявив здатність до самостійного оновлення і поповнення знань.

Оцінка «**Добре**» виставляється студенту, який виявив повне знання навчально-програмного матеріалу, успішно виконує передбачені програмою завдання, засвоїв основну літературу, що рекомендована програмою, показав достатній рівень знань з навчальної дисципліни і здатний до їх самостійного оновлення та поповнення у ході подальшого навчання та професійної діяльності.

Оцінка «**Задовільно**» виставляється студенту, який виявив знання основного навчально-програмного матеріалу в обсязі, необхідному для подальшого навчання та наступної роботи за професією, справляється з виконанням завдань, передбачених програмою, допустив окремі похибки при виконанні екзаменаційних завдань, але володіє необхідними знаннями для подолання допущених похибок під керівництвом педагогічного працівника.

Оцінка «**Незадовільно**» виставляється студенту, який не виявив достатніх знань основного навчально-програмного матеріалу, допустив принципові помилки у виконанні передбачених програмою завдань, не може без допомоги викладача використати знання при подальшому навчанні, не спромігся оволодіти навичками самостійної роботи.