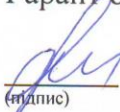


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ АЕРОКОСМІЧНИЙ УНІВЕРСИТЕТ  
ІМ. М.С. ЖУКОВСЬКОГО  
«ХАРКІВСЬКИЙ АВІАЦІЙНИЙ ІНСТИТУТ»

Кафедра ІНФОРМАЦІЙНО-КОМУНІКАЦІЙНИХ ТЕХНОЛОГІЙ  
ІМ. О. О. ЗЕЛЕНСЬКОГО № 504

**“ЗАТВЕРДЖУЮ”**

Гарант освітньої програми

  
(підпис) Ірина ВАСИЛЬСВА  
(ініціали та прізвище)

31 серпня 2023 р.

**РОБОЧА ПРОГРАМА *ОБОВ'ЯЗКОВОЇ*  
НАВЧАЛЬНОЇ ДИСЦИПЛІНИ**

«Об'єктно-орієнтоване програмування та ефективні практики»  
(назва навчальної дисципліни)

Галузь знань: 17 «Електроніка та телекомунікації»  
(шифр і найменування галузі знань)

Спеціальність: 172 «Телекомунікації та радіотехніка»  
(код і найменування спеціальності)

Освітня програма: «Інжиніринг і програмування інфокомунікаційних систем»  
(найменування освітньої програми)

**Форма навчання: денна**

**Рівень вищої освіти: перший (бакалаврський)**

**Харків 2023 рік**

Розробник: РУБЕЛЬ Олексій, доцент, к.т.н.  
(прізвище та ініціали, посада, науковий ступінь і вчене звання)

  
(підпис)

Робочу програму розглянуто на засіданні кафедри інформаційно-комунікаційних технологій ім. О. О. Зеленського №504  
(назва кафедри)

Протокол № 1 від 31 серпня 2023 р.

Завідувач кафедри д.т.н., професор  
(науковий ступінь і вчене звання)

  
(підпис)

Володимир ЛУКІН  
(ініціали та прізвище)

## 1. Опис навчальної дисципліни

Найменування показника	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни (денна форма навчання)
Кількість кредитів – <b>4</b>	<p><b>Галузь знань</b> 17 «<u>Електроніка та телекомунікації</u>» (шифр і найменування)</p> <p><b>Спеціальність</b> 172 «<u>Телекомунікації та радіотехніка</u>» (код і найменування)</p> <p><b>Освітня програма</b> «<u>Інжиніринг і програмування інфокомунікаційних систем</u>» (найменування)</p> <p><b>Рівень вищої освіти:</b> перший (бакалаврський)</p>	Обов'язкова
Кількість модулів – <b>1</b>		<b>Навчальний рік</b>
Кількість змістовних модулів – <b>2</b>		2023/2024
Індивідуальне завдання – <b>не передбачене навчальним планом</b>		<b>Семестр</b>
Загальна кількість годин – <b>56* / 120</b>		3-й (1-й**)
Кількість тижневих годин для денної форми навчання: аудиторних – <b>3,5</b> самостійної роботи здобувача – <b>4</b>		<b>Лекції*</b>
		<u>32</u> години
		<b>Практичні*</b>
		<u>24</u> годин
		<b>Лабораторні*</b>
	-	
	<b>Самостійна робота</b>	
	<u>64</u> годин	
<b>Вид контролю</b>		
модульний контроль, залік		

Співвідношення кількості годин аудиторних занять до самостійної роботи становить: **56 / 64.**

\* Аудиторне навантаження може бути зменшене або збільшене на одну годину залежно від розкладу занять.

\*\* група зі скороченим терміном навчання

## 2. Мета та завдання навчальної дисципліни

**Мета вивчення:** набуття студентами знань і вмінь, використання їх у своїй практичній роботі, пов'язаній з розробкою програмного забезпечення за допомогою мови програмування Java, використанням усіх елементів об'єктно-орієнтованого програмування у ході розробки програмного забезпечення інформаційно-комунікаційних систем і сервісів.

**Завдання:** вивчення сучасних засобів розробки програмного забезпечення, мови програмування Java, об'єктно-орієнтованого програмування.

### **Компетентності, які набуваються:**

#### **загальні:**

- (ЗК1) здатність до абстрактного мислення, аналізу та синтезу;
- (ЗК2) здатність застосовувати знання у практичних ситуаціях;
- (ЗК3) здатність планувати та управляти часом;
- (ЗК5) здатність спілкуватися державною мовою як усно, так і письмово;
- (ЗК7) здатність вчитися і оволодівати сучасними знаннями;
- (ЗК8) вміння виявляти, ставити та вирішувати проблеми;

#### **спеціальні (фахові):**

- (СК1) здатність розуміти сутність і значення інформації в розвитку сучасного інформаційного суспільства;
- (СК3) здатність використовувати базові методи, способи та засоби отримання, передавання, обробки та зберігання інформації.

### **Очікувані результати навчання:**

- (ПРН3) вміння застосовувати знання в галузі інформатики й сучасних інформаційних технологій, обчислювальної і мікропроцесорної техніки та програмування, програмних засобів для розв'язання спеціалізованих задач та практичних проблем у галузі професійної діяльності;
- (ПРН4) здатність брати участь у створенні прикладного програмного забезпечення для елементів (модулів, блоків, вузлів) телекомунікаційних систем, інфокомунікаційних, телекомунікаційних мереж, радіотехнічних систем та систем телевізійного й радіомовлення тощо;
- (ПРН7) здатність брати участь у проектуванні нових (модернізації існуючих) телекомунікаційних систем, інфокомунікаційних, телекомунікаційних мереж, радіотехнічних систем та систем телевізійного й радіомовлення тощо.

**Пререквізити** – «Алгоритми і структури даних», «Основи програмування».

**Кореквізити** – «Front-end програмування», «Back-end програмування», «Програмування мобільних додатків».

### 3. Зміст навчальної дисципліни

#### Модуль 1.

##### Змістовний модуль 1. Основи мови програмування Java

**Тема 1. Java і віртуальна машина Java.** Предмет вивчення і задачі дисципліни. Місце дисципліни в навчальному плані. Необхідне програмне забезпечення. Віртуальна машина Java (JVM). Java Runtime Environment (JRE). Java Development Kit (JDK). Java SE. Компоненти JVM. Загальна архітектура JVM. Платформа Java. Як писати і запускати програми в середовищі Java. Компіляція програми. Java компілятор javac. Запуск програм. Інструменти для розробки на Java. IntelliJ IDEA. Установка середовища розробки JetBrains IntelliJ IDEA.

**Тема 2. Основи Java.** Створення нових проектів в IntelliJ IDEA і налагоджування середовища розробки. Конфігурації запуску IntelliJ IDEA. Створення першого проекту. Інструменти збірки програм. Архів Java (JAR файли). Огляд Gradle. Створення програм за допомогою Gradle. Вступ до Java. Основні особливості Java. Деякі загальні правила для Java. Основні синтаксичні особливості мови програмування Java. Основні літерали: числа, рядки і символи. Змінні і значення. Написання першої програми. Друк даних – println() і print(). Читання введених користувачем даних за допомогою сканера. Scanner. Читання багаторядкового введення. Форматований вивід даних. Методи System.out.print() і System.out.println(). Метод printf(). Метод String.format().

**Тема 3. Типи даних і змінні.** Типи даних і змінні. Примітивні типи даних. Доступ до значення змінної. Висновок типу (type inference). Типи даних і їх розміри. Числа. Цілі числа. Числа з плаваючою комою – float і double. Читання чисел з плаваючою комою. Десятковий роздільник. Символи. Булеві типи. Ключове слово new. Перетворення типів – неявне та явне. Примітивні (primitive) типи та типи посилань (reference). Ключове слово new. Зберігання даних у пам'яті. Два основних простори пам'яті: стек і купа. Тип null. Константи. Кінцеві змінні. Ключове слово final. Коли використовувати кінцеві змінні.

**Тема 4. Операції над примітивними типами.** Операції над цілими числами. Форми оператора присвоєння. Читання чисел зі стандартного введення. Інкремент і декремент. Арифметичні операції. Порівняння значень. Оператори відношення. Пріоритет операторів. Символи і escape-последовності. Булев тип і операції з ним. Логічні оператори (NOT, AND, OR і XOR). Пріоритет логічних операторів.

**Тема 5. Керуючі оператори потоку.** Умовні оператори. Випадки використання if, if-else, if-else-if. Вкладені конструкції. Тернарний оператор. Оператор switch, case, default. Використання switch як вираз. Цикли. Цикл for. Вкладені цикли. Цикл while. Цикл do-while. Читання послідовності невідомої довжини. Розгалуження. Оператори переходу – break, continue, return.

**Тема 6. Масиви і рядки.** Масиви в Java. Оголошення і ініціалізація масивів. Створення масиву із заданими елементами. Створення масиву за допомогою ключового слова "new". Довжина масиву. Доступ до елементів. Ітерація по масивах. Обробка масивів за допомогою циклів. Читання масиву зі стандартного введення. Використання вбудованих методів і функцій для роботи з масивами. Тип String. Створення рядків. Довжину та символи рядка. Корисні методи роботи з рядками. Конкатенація рядків. Додавання значень до рядка. Порівняння рядків. Обробка рядків. Метод split. Ітерація по рядку.

##### Тема 7. Стиль коду.

Коментарі. Багаторядкові коментарі. Іменування змінних. Правила іменування змінних. Положення про стиль написання коду. Конвенції Java. Кількість пробілів. Довжина рядка.

#### Модульний контроль

##### Змістовний модуль 2. Об'єктно-орієнтоване програмування в Java

**Тема 1. Основи об'єктно-орієнтованого програмування.** Парадигми програмування. Об'єктно-орієнтоване програмування (ООП). Визначення ООП. Ознайомлення з поняттями класів, об'єктів, атрибутів і методів у Java. Принципи ООП: успадкування, поліморфізм, інкапсуляція, абстракція. Основні відомості про класи. Оголошення класів. Константи класу.

Властивості класів. Атрибути (змінні екземпляра і класу). Методи класу. Оголошення методів. Синтаксис методів. Метод new. Модифікатори методів. Метод main. Статичні методи (static), методи класу і екземпляра класу. Рекурсивні функції. Перевантаження методу. Передача масивів методам. Аргументи командного рядка. Методи екземплярів (instance). Порядок створення екземплярів класів. Ініціалізація нових екземплярів. Виклик методу. Конструктор. Перевантаження конструктора. Виклик конструкторів з інших конструкторів. Поняття про об'єкти і їх використання. Методи доступу: гетери і сетери. Зміна атрибутів, пов'язаних з об'єктом через методи. Типові і параметризовані конструктори. Архітектура програмного забезпечення. Застосування UML для моделювання об'єктно-орієнтованих систем. Діаграми UML. UML діаграма компонентів.

**Тема 2. Класи і особливості використання ООП в Java.** Різновиди класів в Java. Організація класів у пакети. Імпортування класів. Модулі. Модифікатори доступу: public, protected, private, package-private (default). Документування класів і пакетів javadoc. Перевантаження конструкторів. Ієрархія класів. Особливості успадкування в Java. Доступ до полів і методів суперкласу. Ключове слово "super". Виклик конструктора суперкласу. Посилання на об'єкти підкласу. Доступ до полів і методів через посилання суперкласу. Ключове слово instanceof. Поліморфізм в Java. Поліморфізм в ієрархії класів. Перевизначення методів. Правила перевизначення методів. Заборона перевизначення. Перевизначення та перевантаження методів разом. Приховування статичних методів. Внутрішні класи. Вкладені класи. Анонімний клас. Класи-перерахування enum. Визначення enum. Методи обробки enum. Поля і методи в enum. Абстрактний клас. Інтерфейси. Відмінності між абстрактними класами і інтерфейсами. Оголошення інтерфейсів. Спільне використання абстрактних класів і інтерфейсів. Реалізація інтерфейсів. Реалізація і розширення декількох інтерфейсів. Статичні методи в інтерфейсах. Лямбда-вирази і функціональні інтерфейси. Анотації. Додавання анотацій. Користувальницькі анотації і типи анотацій. UML діаграма класів. Файли класів і байт-код. Компіляція вихідного файлу. Стек викликів і його структура. Стек і виконання методів. Переповнення стека. Трасування стека. Огляд класу StackTraceElement.

**Тема 3. Виключення і їх обробка.** Помилки під час компіляції і помилки під час виконання. Винятки. Ієрархія винятків. Базовим клас всіх винятків java.lang.Throwable. Створення винятків. Оператор throw. Обробка винятків. Оператор try-catch. Перехоплення кількох винятків. Блок finally. NullPointerException (NPE). Правила уникнення NPE. Створення власних підкласів виключень.

**Тема 4. Робота з даними.** Клас File. Основні методи. Читання файлів. Запис файлів. Клас FileWriter. Клас PrintWriter. Керування файлами. Створення файлів. Регулярні вирази в Java. Класи java.util.regex.Pattern і java.util.regex.Matcher. Огляд класів Math, LocalDate, LocalTime і LocalDateTime.

**Тема 5. Колекції і огляд фреймворку Collections.** Колекції. Змінні і незмінні колекції. Клас ArrayList і його методи. EnumSet. Інтерфейс Collections. Інтерфейс Set. Інтерфейс List. Інтерфейс Map. Стек. Дек. Розгляд класів String, StringBuffer, StringBuilder.

**Тема 6. Основи шаблонів проектування.** Шаблони проектування. Принципи SOLID. Творчі, структурні і поведінкові шаблони проектування. Шаблон Singleton. Шаблон фабричний метод (factory method). Шаблон фасад (facade). Шаблон адаптер. Шаблон декоратор. Шаблон ланцюжок відповідальностей. Шаблони проектування MVC і MVP. Чиста архітектура (Clean architecture) як підхід до організації архітектури програмного забезпечення.

**Тема 7. Основи багатопоточності в Java.** Що таке потоки. Потокова модель в Java. Потоки як об'єкти. Клас потоків java.lang.Thread. Головний потік (main). Вхідні і вихідні потоки. Потоки байтів і символів. Буферизовані потоки. Створення власних потоків. Запуск потоків. Проста багатопотокова програма. Керування потоками: призупинення, відновлення і зупинка потоків. Винятки в потоках. Стани потоку – Thread.State enum клас. Життєвий цикл потоку. Обмін даними між потоками. Пріоритети потоків. Синхронізація потоків. Об'єднання і синхронізація потоків: ключове слово synchronized і бібліотека java.util.concurrent.

**Тема 8. Інструменти тестування і налагодження програм.** Налагодження (debugging) простих конструкцій. Використання assert. Точки зупинки конструктора і методу. Точки зупинки винятків. Покрове налагодження (stepping). Вступ до логування в Java (Java Logging). Клас Logger. Logback. Логування/'printf' налагодження. Основи модульного тестування. Модульне тестування з JUnit. Написання тестів. Клас Assertions фреймворку JUnit. Запуск тестів. Результати тестування.

**Модульний контроль**

**4. Структура навчальної дисципліни**

Назва змістовного модуля і тем	Кількість годин				
	Усього	У тому числі			
		л	п	лаб.	с. р.
1	2	3	4	5	6
<b>Модуль 1</b>					
<b>Змістовний модуль 1. Основи мови програмування Kotlin</b>					
Тема 1. Java і віртуальна машина Java	6	2	1	-	3
Тема 2. Основи Java	5	1	1	-	3
Тема 3. Типи даних і змінні	9	2	2	-	5
Тема 4. Операції над примітивними типами	8	1	2	-	5
Тема 5. Керуючі оператори потоку	6	1	1	-	4
Тема 6. Масиви і рядки	7	1	2	-	4
Тема 7. Стиль коду	5	1	1	-	3
<b>Модульний контроль</b>	3	1	-	-	2
Разом за змістовним модулем 1	49	10	10		29
<b>Змістовний модуль 2. Об'єктно-орієнтоване і функціональне програмування в Kotlin</b>					
Тема 1. Основи об'єктно-орієнтованого програмування	10	4	2	-	4
Тема 2. Класи і особливості використання ООП в Java	13	6	2	-	5
Тема 3. Виключення і їх обробка	7	1	2	-	4
Тема 4. Робота з даними	7	1	2	-	4
Тема 5. Колекції і огляд фреймворку Collections	8	2	2	-	4
Тема 6. Основи шаблонів проектування	10	4	2	-	4
Тема 7. Основи багатопоточності в Java.	7	2	1	-	4
Тема 8. Інструменти тестування і налагодження програм	6	1	1	-	4
<b>Модульний контроль</b>	3	1	-	-	2
Разом за змістовним модулем 2	71	22	14	-	35
<b>Усього годин</b>	120	32	24	-	64

## 5. Теми семінарських занять

№ п/п	Назва теми	Кількість годин
1	Не передбачено навчальним планом	
	<b>Разом</b>	

## 6. Теми практичних занять

№ п/п	Назва теми	Кількість годин
1	Установка Java Development Kit. Перша програма	2
2	Робота з консоллю	2
3	Проект з роботою з текстом і операціями вводу/виводу	2
4	Реалізація гри «Хрестики-Нулики»	4
5	Обробка виключень	2
6	Розробки багатофункціонального калькулятора	4
7	Використанням об'єктно-орієнтованого програмування у створенні складних моделей	4
8	Робота з колекціями	2
9	Робота з потоками	1
10	Модульне тестування з JUnit	1
	<b>Разом</b>	24

## 7. Теми лабораторних занять

№ п/п	Назва теми	Кількість годин
1	Не передбачено навчальним планом	
	<b>Разом</b>	

## 8. Самостійна робота

№ п/п	Назва теми	Кількість годин
1	Опрацювання матеріалу лекцій	32
2	Підготовка до практичних робіт	16
3	Опрацювання матеріалів та результатів отриманих на практичних заняттях	12
4	Підготовка до модульного контролю	4
	<b>Разом</b>	64

## 9. Індивідуальні завдання

Не передбачено навчальним планом.



## 10. Методи навчання

При викладанні курсу використовуються наступні навчальні методи:

- демонстрація;
- ілюстрація;
- розповідь;
- спостереження;
- дослідження;
- практична робота.

## 11. Методи контролю

Для контролю успішності в даному курсі використано:

- поточний контроль на практичних заняттях;
- модульний контроль за змістовними модулями;
- семестровий контроль у вигляді заліку.

## 12. Критерії оцінювання та розподіл балів, які отримують здобувачі

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
<b>Змістовний модуль 1</b>			
Виконання і захист практичних робіт	0...5	6	0...30
Модульний контроль	0...20	1	0...20
<b>Змістовний модуль 2</b>			
Виконання і захист практичних робіт	0...5	6	0...30
Модульний контроль	0...20	1	0...20
<b>Усього за семестр</b>			<b>0...100</b>

Семестровий контроль проводиться у разі відмови студента від балів поточного контролю і за наявності допуску до заліку.

Білет для заліку складається з тридцяти тестових теоретичних і практичних питань. Максимальна сума балів – 100 балів.

### Критерії оцінювання роботи здобувача протягом семестру

**Задовільно (60-74).** Показати мінімум знань та умінь. Захистити всі практичні завдання і здати тестування. Знати синтаксис мови програмування Java і базові принципи програмування. Мати уявлення про об'єктно-орієнтоване програмування.

**Добре (75-89).** Твердо знати мінімум, захистити всі практичні завдання, здати тестування і поза аудиторну самостійну роботу. Уміти: окрім наведених базових знань застосовувати шаблони проектування для написання гнучкого і придатного для масштабування програмного забезпечення, знати основні елементи і прийоми об'єктно-орієнтованого програмування.

**Відмінно (90-100).** Здати всі контрольні точки з оцінкою «відмінно». Досконально знати всі теми та уміти застосовувати їх.

### Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

### 13. Методичне забезпечення

1. Скоб Ю.О. Основи програмування мовою JAVA: навч. посіб. до лаб. практикуму / Ю. О. Скоб, М. Л. Угрюмов, В. О. Халтурін ; М-во освіти і науки України, Національний аерокосмічний університет ім. М. Є. Жуковського "Харківський авіаційний інститут". - Харків. - Національний аерокосмічний університет ім. М. Є. Жуковського " Харківський авіаційний інститут ", 2017. - 112 с.

### 14. Рекомендована література

#### Базова

1. Sierra K. Head First Java: A Brain-Friendly Guide, 3<sup>rd</sup> Edition / K. Sierra, B. Bates, Trisha Gee. – O'Reilly Media, 2022. – 752 p.
2. Downey A., Mayfield C. Think Java: How to Think Like a Computer Scientist / A. Downey, C. Mayfield. – O'Reilly Media, 2016. – 249 p.
3. Schildt H. Java: A Beginner's Guide, Eighth Edition 8th Edition / H. Schildt. – McGraw Hill, 2018. – 720 p.
4. Schildt H. Java: The Complete Reference, Twelfth Edition 12th Edition / H. Schildt. – McGraw Hill, 2021. – 1280 p.

#### Допоміжна

1. Bloch J. Effective Java 3rd Edition / J. Bloch. – Addison-Wesley Professional, 2017. – 416 p.
2. Niemeyer P., Leuck D. Learning Java: A Bestselling Hands-On Java Tutorial 4th Edition / P. Niemeyer, D. Leuck. – O'Reilly Media, 2013. – 1007 p.
3. Martin R. Clean Code: A Handbook of Agile Software Craftsmanship / Robert C. Martin. – US: Pearson Education, 2008. – 464 p.
4. Martin R. Clean Architecture: A Craftsman's Guide to Software Structure and Design / Robert C. Martin. – US: Pearson, 2017. – 432 p.
5. Freeman E., Robson E. Head First Design Patterns: Building Extensible and Maintainable Object-Oriented Software / E. Freeman, E. Robson. – O'Reilly Media, 2021. – 669 p.

### 15. Інформаційні ресурси

1. Науково-технічна бібліотека ХАІ - Режим доступу: <http://library.khai.edu>
2. Навчальні матеріали з Java - <https://docs.oracle.com/javase/tutorial/tutorialLearningPaths.html>
3. Інтерактивний підручник з Java - <https://www.learnjavaonline.org/>
4. Посібник зі стилю для коду Java - <https://google.github.io/styleguide/javaguide.html#s6.1-override-annotation>