

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСЛКИ  
«ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ»  
Навчально-науковий інститут денної освіти  
Кафедра комп'ютерних наук та інформаційних технологій**

ЗАТВЕРДЖУЮ

Завідувач кафедри КНІТ

 Олена ОЛЬХОВСЬКА

«25» січня 2023 р.

**РОБОЧА ПРОГРАМА**

навчальної дисципліни  
освітня програма  
спеціальність  
галузь знань  
ступінь вищої освіти

**«Програмування І»  
Комп'ютерні науки  
122 Комп'ютерні науки  
12 Інформаційні технології  
бакалавр**


Робоча програма навчальної дисципліни «Програмування І» рекомендована до використання в освітньому процесі на засіданні кафедри комп'ютерних наук та інформаційних технологій  
Протокол від 25 січня 2023 року, №8

**Полтава 2023**

Укладач: Олексійчук Юрій Федорович, доцент кафедри комп'ютерних наук та інформаційних технологій, к. ф.-м. н.

ПОГОДЖЕНО:

Гарант освітньої програми «Комп'ютерні науки»  
спеціальності 122 "Комп'ютерні науки"  
ступеня бакалавр

  
\_\_\_\_\_ Оксана ЧЕРНЕНКО  
підпис ініціали, прізвище

«25» січня 2023 року

## **Зміст** **робочої програми початкової дисципліни**

Розділ 1. Загальна характеристика навчальної дисципліни	4
Розділ 2. Перелік компетентностей та програмні результати навчання з навчальної дисципліни	4
Розділ 3. Програма навчальної дисципліни	6
Розділ 4. Тематичний план навчальної дисципліни	6
Розділ 5. Оцінювання результатів навчання	14
Розділ 6. Інформаційні джерела	16
Розділ 7. Програмне забезпечення навчальної дисципліни	16

## Розділ 1. Загальна характеристика навчальної дисципліни

Таблиця 1. Загальна характеристика навчальної дисципліни "Програмування І"

Місце у структурно-логічній схемі підготовки	<i>Пререквізити:</i> –  <i>Постреквізити:</i> Платформи корпоративних інформаційних систем, Сучасні парадигми програмування
Мова викладання	Українська
Статус дисципліни	Обов'язкова
Курс/семестр вивчення	1/1-2
Кількість кредитів ECTS/кількість модулів	8/3
Денна форма навчання:	
Кількість годин: 240 год – загальна кількість: 1 семестр – 120 год, 2 семестр – 120 год	
- Лекції: 48 год., 1 семестр – 24 год, 2 семестр – 24 год	
- Практичні (семінарські, лабораторні) заняття: 48 год., 1 семестр – 24 год, 2 семестр – 24 год	
- Самостійна робота: 144 год., 1 семестр – 72 год, 2 семестр – 72 год.	
- Вид підсумкового контролю (ПМК, екзамен): 1 семестр - ПМК, 2 семестр – екзамен	
Заочна форма навчання:	
Кількість годин: 240 год – загальна кількість: 1 семестр – 120 год, 2 семестр – 120 год	
- Лекції: 12 год., 1 семестр – 6 год, 2 семестр – 6 год	
- Практичні (семінарські, лабораторні) заняття: 4 год., 1 семестр – 2 год, 2 семестр – 2 год	
- Самостійна робота: 224 год., 1 семестр – 112 год, 2 семестр – 112 год	
- Вид підсумкового контролю (ПМК, екзамен): 1 семестр - ПМК, 2 семестр – екзамен	

## Розділ 2. Перелік компетентностей та програмні результати навчання з навчальної дисципліни

Основною метою вивчення дисципліни «Програмування І» є формування у студентів системного мислення та навичок алгоритмічного програмування та об'єктно-орієнтованого програмування.

Головним завданням дисципліни є набуття студентами знань та вмінь програмування, навчитися працювати з основними засобами програмування.

Предметом навчальної дисципліни «Програмування І» є процес створення алгоритмів та програм.

Таблиця 2. Перелік компетентностей та програмні результати навчання, які забезпечує навчальна дисципліна «Програмування І»

Програмні результати навчання	Компетентності, якими повинен оволодіти здобувач
<p>ПР1. Застосовувати знання основних форм і законів абстрактно-логічного мислення, основ методології наукового пізнання, форм і методів вилучення, аналізу, обробки та синтезу інформації в предметній області комп'ютерних наук.</p> <p>ПР5. Проектувати, розробляти та аналізувати алгоритми розв'язання обчислювальних та логічних задач, оцінювати ефективність та складність алгоритмів на основі застосування формальних моделей алгоритмів та обчислюваних функцій.</p>	<p>Здатність до абстрактного мислення, аналізу та синтезу (ЗК1).</p> <p>Здатність застосовувати знання у практичних ситуаціях (ЗК2).</p> <p>Знання та розуміння предметної області та розуміння професійної діяльності (ЗК3).</p> <p>Здатність спілкуватися державною мовою як усно, так і письмово (ЗК4).</p> <p>Здатність вчитися й оволодівати сучасними знаннями (ЗК6).</p> <p>Здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК7).</p> <p>Здатність генерувати нові ідеї (креативність) (ЗК8).</p> <p>Здатність працювати в команді (ЗК9).</p> <p>Здатність бути критичним і самокритичним (ЗК10).</p> <p>Здатність приймати обґрунтовані рішення (ЗК11)</p> <p>Здатність оцінювати та забезпечувати якість виконуваних робіт (ЗК12).</p> <p>Здатність діяти на основі етичних міркувань (ЗК13).</p> <p>Здатність до математичного формулювання та досліджування неперервних та дискретних математичних моделей, обґрунтування вибору методів і підходів для розв'язування теоретичних і прикладних задач у галузі комп'ютерних наук, аналізу та інтерпретування (СК1).</p> <p>Здатність до логічного мислення, побудови логічних висновків, використання формальних мов і моделей алгоритмічних обчислень, проектування, розроблення й аналізу алгоритмів, оцінювання їх ефективності та складності, розв'язності та нерозв'язності алгоритмічних проблем для адекватного моделювання предметних областей і створення програмних та інформаційних систем (СК3).</p>

### Розділ 3. Програма навчальної дисципліни

#### Модуль 1 «Основи програмування алгоритмічною мовою»

Тема 1. Інструменти і базові засоби програмування.

Мови програмування. Інтегровані середовища розробки. Особливості мови програмування Java.

Тема 2. Команди та дані. Структури керування.

Оператори. Типи даних. Цикли. Галуження. Масиви. Методи. Оператори керування.

#### Модуль 2. «Основи об'єктно-орієнтованого програмування»

Тема 3. ООП в Java.

Об'єкти. Класи. Інтерфейси. Успадкування. Методи по замовчуванню в інтерфейсах. Статичні поля. Статичні методи. Перевантаження. Перевизначення. Клас Object. Модифікатори доступу.

#### Модуль 3 «Основні концепції алгоритмічних мов»

Тема 4. Абстракція даних. Складені структури даних.

Структури даних. Колекції в Java. Списки. Черги. Множини. Мапи.

Тема 5. Алгоритмічна декомпозиція

Робота з файлами. Maven. Декомпозиція. Графічний інтерфейс користувача.

### Розділ 4. Тематичний план навчальної дисципліни

Таблиця 3. Тематичний план навчальної дисципліни для студентів денної форми навчання

1	2	3	4	5	6
Назва теми (лекції) та питання теми (лекції)	кількість годин	Назва теми та питання семінарського, практичного або лабораторного заняття	кількість годин	Завдання самостійної роботи в розрізі тем	кількість годин
<b>Модуль 1 «Основи програмування алгоритмічною мовою»</b>					
Тема 1. Інструменти і базові засоби програмування. Лекція 1. Вступ до програмування. 1. Вступ. 2. Огляд основних мов	2	Практичне заняття 1. Знайомство з середовищем розробки NetBeans	2	опрацювати лекційний матеріал до теми 1, готуватись до практичних занять, пройти	6

1	2	3	4	5	6
програмування. 3. Поняття парадигми програмування.				тести до лекцій в ДК Опрацювати теми: 1. Онлайн компілятори. 2. Інтегровані середовища розробки.	
Тема 2. Команди та дані. Структури керування. Лекція 2. Основи мови програмування Java 1. Перша програма 2. Основні операції 3. Прості типи даних в Java 4. Перетворення типів 5. Масиви Лекція 3. Основні конструкції керування в Java 1. Оператори вибору 2. Оператори циклу 3. Оператори переходу. 4. Приклади Лекція 4. Масиви 1. Одновимірні масиви 2. Багатовимірні масиви Лекція 5. Введення даних 1. Клас Scanner. 2. Робота з об'єктом класу Scanner. 3. Приклади. Лекція 6. Методи 1. Методи 2. Перевантаження методів 7. Рекурсія	2     2     2     2	Практичне заняття 2. Робота з масивами  Практичне заняття 3. Основні оператори  Практичне заняття 4. Цикли. Двовимірні масиви  Практичне заняття 5. Введення даних. Оператори вибору  Практичне заняття 6. Рекурсія  Практичне заняття 7. Методи  Практичне заняття 8. ПМР. Структури керування	2  2  2  2  2  2	опрацювати лекційний матеріал до теми 2, готуватись до практичних занять, пройти тести до лекцій в ДК Опрацювати теми: 1. Побітові операції. 2. Лямбда-вирази. 3. Двійкове представлення даних.	36
<b>Модуль 2 «Основи об'єктно-орієнтованого програмування»</b>					
Тема 3. ООП в Java. Лекція 7. Структура Java-програм. 1. Поняття класу 2. Створення класів в NetBeans 3. Структура файлів проекту 4. Методи 5. Конструктори Лекція 8. Робота з	2      2	Практичне заняття 9. Успадкування  Практичне заняття 10. Перевантаження  Практичне заняття 11. Абстрактні класи та інтерфейси	2  2  2	опрацювати лекційний матеріал до теми 3, готуватись до практичних занять, пройти тести до лекцій в ДК Опрацювати теми: 1. Порівняння	30

1	2	3	4	5	6
<p>об'єктами</p> <p>1. Відмінність процедурних від об'єктно-орієнтованих програм</p> <p>2. Змінні-значення і змінні-посилання</p> <p>3. Об'єкти і збір сміття</p> <p>Лекція 9. Спадкування в Java</p> <p>1. Основи спадкування.</p> <p>2. Ключові слова this та super.</p> <p>3. Особливості роботи з методами та конструкторами.</p> <p>4. Абстрактні класи.</p> <p>Лекція 10. Пакети та інтерфейси в Java</p> <p>1. Пакети</p> <p>2. Модифікатори доступу</p> <p>3. Інтерфейси</p> <p>4. Інтерфейсні змінні</p> <p>5. Спадкування інтерфейсів</p> <p>6. Вкладені інтерфейси</p> <p>7. Методи по замовчуванню</p> <p>Лекція 11. Обробка виключних ситуацій</p> <p>1. Виключні ситуації</p> <p>2. Типи виключень</p> <p>3. Робота з виключними ситуаціями</p> <p>4. Вбудовані виключення Java</p> <p>Лекція 12. Клас Object</p> <p>1. Методи класу Object.</p> <p>2. Метод clone()</p> <p>3. Незмінні об'єкти</p>	2	<p>Практичне заняття 12. ПМР. Основи ООП</p>	2	<p>ООП в Java з іншими мовами програмування.</p> <p>2. Анотації</p>	
<b>Модуль 3 «Основні концепції алгоритмічних мов»</b>					
<p>Тема 4. Абстракція даних. Складені структури даних.</p> <p>Лекція 13. Вступ до структур даних</p> <p>1. Поняття про структури даних.</p> <p>2. Приклад структур даних.</p>	2	<p>Практичне заняття 13. Створення прикладних програм з графічним інтерфейсом</p>	2	<p>опрацювати лекційний матеріал до теми 4, готуватись до практичних занять, пройти тести до теми в ДК</p>	36
	2	<p>Практичне заняття 14. Робота з ArrayList</p>	2	<p>Опрацювати теми:</p>	



1	2	3	4	5	6
Лекція 14. Вступ до структур даних. Колекції 3. Колекції Java 4. Інтерфейс Collection 5. Інтерфейс List 6. Стандартні класи колекцій 7. Клас ArrayList Лекція 15. Черги і зв'язані списки 1. Інтерфейс Queue. 2. Інтерфейс Deque. 3. Клас ArrayDeque. 4. Клас LinkedList. 5. Приклади. Лекція 16. Множини (набори) 1. Інтерфейси Set та SortedSet 2. Інтерфейс NavigableSet 3. Клас HashSet 4. Клас LinkedHashSet 5. Клас TreeSet 6. Компаратори 7. Приклади Лекція 17. Карти 1. Ітератори. 2. Карти. 3. Інтерфейси Map. 4. Класи Map. 5. Переліки Лекція 18. Інші класи та інтерфейси java.util 1. Інтерфейс Enumeration 2. Клас Vector 3. Клас Stack 4. Класи Dictionary та Hashtable 5. Клас Properties	2    2   2   2	Практичне заняття 15. ArrayDeque  Практичне заняття 16. LinkedList  Практичне заняття 17. Генерація псевдовипадкових чисел. Множини  Практичне заняття 18. Компаратори  Практичне заняття 19. ПМР. Колекції.	2   2   2   2	1. Червоно-чорні дерева. 2. Асимптотичні оцінки функцій	
Тема 5. Алгоритмічна декомпозиція Лекція 19. Алгоритмічна декомпозиція 1. Модульна структура програм. 2. Декомпозиція. 3. Робота з бібліотеками. Лекція 20. Документування. Вимоги до коду 1. Javadoc 2. Дескриптори	2   2	Практичне заняття 20. Робота з файлами  Практичне заняття 21. Обробка тексту  Практичне заняття 22. Графічний інтерфейс користувача  Практичне заняття 23.	2  2  2  2	опрацювати лекційний матеріал до теми 5, готуватись до практичних занять, пройти тести до теми в ДК Опрацювати теми: 1. Автоматична збірка. 2. Патерни	36

1	2	3	4	5	6
3. Вимоги до коду Лекція 21. Графічний інтерфейс користувача. 1. AWT і Swing 2. Клас Component 3. Контейнери 4. Клас Panel 5. Класи Window і Frame Лекція 22. Відношення між об'єктами (класами) 1. Основні типи відношень 2. Асоціація 3. Агрегація і композиція Лекція 23. SOLID 1.SOLID 2. Принцип єдиного обов'язку 3. Принцип відкритості/закритості 4. Принцип підстановки Барбара Лісков 5. Принцип розділення інтерфейсу 6. Принцип інверсії залежностей Лекція 24. Еволюція Java 1. Історія Java 2. Версії Java 3. Лямбда-вирази 4. Посилання на методи	2 2 2 2	Графічний інтерфейс користувача  Практичне заняття 24. ПМР. Алгоритмічна декомпозиція	2	проектування.	
<b>Всього, годин</b>	<b>48</b>		<b>48</b>		<b>144</b>

Таблиця 4. Тематичний план навчальної дисципліни для студентів заочної форми навчання

1	2	3	4	5	6
Назва теми (лекції) та питання теми (лекції)	кількість годин	Назва теми та питання семінарського, практичного або лабораторного заняття	кількість годин	Завдання самостійної роботи в розрізі тем	кількість годин
<b>Модуль 1 «Основи програмування алгоритмічною мовою»</b>					
Тема 1. Інструменти і базові засоби програмування. Лекція 1. Вступ до програмування. 1. Вступ.		Практичне заняття 1. Знайомство з середовищем розробки NetBeans		опрацювати лекційний матеріал до теми 1, готуватись до практичних	10

1	2	3	4	5	6
2. Огляд основних мов програмування. 3. Поняття парадигми програмування.				заняць, пройти тести до лекцій в ДК Опрацювати теми: 1. Онлайн компілятори. 2. Інтегровані середовища розробки.	
Тема 2. Команди та дані. Структури керування. Лекція 2. Основи мови програмування Java 1. Перша програма 2. Основні операції 3. Прості типи даних в Java 4. Перетворення типів 5. Масиви Лекція 3. Основні конструкції керування в Java 1. Оператори вибору 2. Оператори циклу 3. Оператори переходу. 4. Приклади Лекція 4. Масиви 1. Одновимірні масиви 2. Багатовимірні масиви Лекція 5. Введення даних 1. Клас Scanner. 2. Робота з об'єктом класу Scanner. 3. Приклади. Лекція 6. Методи 1. Методи 2. Перевантаження методів 7. Рекурсія	2	Практичне заняття 2. Робота з масивами  Практичне заняття 3. Основні оператори  Практичне заняття 4. Цикли. Двовимірні масиви  Практичне заняття 5. Введення даних. Оператори вибору  Практичне заняття 6. Рекурсія  Практичне заняття 7. Методи  Практичне заняття 8. ПМР. Структури керування		опрацювати лекційний матеріал до теми 2, готуватись до практичних занять, пройти тести до лекцій в ДК Опрацювати теми: 1. Побітові операції. 2. Лямбда-вирази. 3. Двійкове представлення даних.	52
<b>Модуль 2 «Основи об'єктно-орієнтованого програмування»</b>					
Тема 3. ООП в Java. Лекція 7. Структура Java-програм. 1. Поняття класу 2. Створення класів в NetBeans 3. Структура файлів проекту 4. Методи 5. Конструктори		Практичне заняття 9. Успадкування  Практичне заняття 10. Перевантаження  Практичне заняття 11. Абстрактні класи та інтерфейси	2	опрацювати лекційний матеріал до теми 3, готуватись до практичних занять, пройти тести до лекцій в ДК Опрацювати теми:	50

1	2	3	4	5	6
<p>Лекція 8. Робота з об'єктами</p> <p>1. Відмінність процедурних від об'єктно-орієнтованих програм</p> <p>2. Змінні-значення і змінні-посилання</p> <p>3. Об'єкти і збір сміття</p> <p>Лекція 9. Спадкування в Java</p> <p>1. Основи спадкування.</p> <p>2. Ключові слова this та super.</p> <p>3. Особливості роботи з методами та конструкторами.</p> <p>4. Абстрактні класи.</p> <p>Лекція 10. Пакети та інтерфейси в Java</p> <p>1. Пакети</p> <p>2. Модифікатори доступу</p> <p>3. Інтерфейси</p> <p>4. Інтерфейсні змінні</p> <p>5. Спадкування інтерфейсів</p> <p>6. Вкладені інтерфейси</p> <p>7. Методи по замовчуванню</p> <p>Лекція 11. Обробка виключних ситуацій</p> <p>1. Виключні ситуації</p> <p>2. Типи виключень</p> <p>3. Робота з виключними ситуаціями</p> <p>4. Вбудовані виключення Java</p> <p>Лекція 12. Клас Object</p> <p>1. Методи класу Object.</p> <p>2. Метод clone()</p> <p>3. Незмінні об'єкти</p>	<p>2</p> <p>2</p>	<p>Практичне заняття 12. ПМР. Основи ООП</p>		<p>1. Порівняння ООП в Java з іншими мовами програмування.</p> <p>2. Анотації</p>	
<b>Модуль 3 «Основні концепції алгоритмічних мов»</b>					
<p>Тема 4. Абстракція даних. Складені структури даних.</p> <p>Лекція 13. Вступ до структур даних</p> <p>1. Поняття про структури даних.</p> <p>2. Приклад структур</p>	<p>2</p> <p>2</p>	<p>Практичне заняття 13. Створення прикладних програм з графічним інтерфейсом</p> <p>Практичне заняття 14. Робота з ArrayList</p>	<p>2</p>	<p>опрацювати лекційний матеріал до теми 4, готуватись до практичних занять, пройти тести до теми в ДК</p>	<p>56</p>

1	2	3	4	5	6
<p>даних. Лекція 14. Вступ до структур даних. Колекції</p> <p>3. Колекції Java 4. Інтерфейс Collection 5. Інтерфейс List 6. Стандартні класи колекцій 7. Клас ArrayList</p> <p>Лекція 15. Черги і зв'язані списки</p> <p>1. Інтерфейс Queue. 2. Інтерфейс Deque. 3. Клас ArrayDeque. 4. Клас LinkedList. 5. Приклади.</p> <p>Лекція 16. Множини (набори)</p> <p>1. Інтерфейси Set та SortedSet 2. Інтерфейс NavigableSet 3. Клас HashSet 4. Клас LinkedHashSet 5. Клас TreeSet 6. Компаратори 7. Приклади</p> <p>Лекція 17. Карти</p> <p>1. Ітератори. 2. Карти. 3. Інтерфейси Map. 4. Класи Map. 5. Переліки</p> <p>Лекція 18. Інші класи та інтерфейси java.util</p> <p>1. Інтерфейс Enumeration 2. Клас Vector 3. Клас Stack 4. Класи Dictionary та Hashtable 5. Клас Properties</p>		<p>Практичне заняття 15. ArrayDeque</p> <p>Практичне заняття 16. LinkedList</p> <p>Практичне заняття 17. Генерація псевдовипадкових чисел. Множини</p> <p>Практичне заняття 18. Компаратори</p> <p>Практичне заняття 19. ПМР. Колекції.</p>		<p>Опрацювати теми:</p> <p>1. Червоно-чорні дерева. 2. Асимптотичні оцінки функцій</p>	
<p>Тема 5. Алгоритмічна декомпозиція</p> <p>Лекція 19. Алгоритмічна декомпозиція</p> <p>1. Модульна структура програм. 2. Декомпозиція. 3. Робота з бібліотеками.</p> <p>Лекція 20. Документування. Вимоги до коду</p> <p>1. Javadoc</p>		<p>Практичне заняття 20. Робота з файлами</p> <p>Практичне заняття 21. Обробка тексту</p> <p>Практичне заняття 22. Графічний інтерфейс користувача</p>		<p>опрацювати лекційний матеріал до теми 5, готуватись до практичних занять, пройти тести до теми в ДК</p> <p>Опрацювати теми:</p> <p>1. Автоматична збірка.</p>	56

1	2	3	4	5	6
2. Дескриптори 3. Вимоги до коду Лекція 21. Графічний інтерфейс користувача. 1. AWT і Swing 2. Клас Component 3. Контейнери 4. Клас Panel 5. Класи Window і Frame Лекція 22. Відношення між об'єктами (класами) 1. Основні типи відношень 2. Асоціація 3. Агрегація і композиція Лекція 23. SOLID 1. SOLID 2. Принцип єдиного обов'язку 3. Принцип відкритості/закритості 4. Принцип підстановки Барбара Лісков 5. Принцип розділення інтерфейсу 6. Принцип інверсії залежностей Лекція 24. Еволюція Java 1. Історія Java 2. Версії Java 3. Лямбда-вирази 4. Посилання на методи	2	Практичне заняття 23. Графічний інтерфейс користувача  Практичне заняття 24. ПМР. Алгоритмічна декомпозиція		2. Патерни проектування.	
<b>Всього, годин</b>	<b>12</b>		<b>4</b>		<b>224</b>

### Розділ 5. Оцінювання результатів навчання

Таблиця 5. Розподіл балів за результатами вивчення навчальної дисципліни

1 семестр

Форма навчальної роботи	Вид навчальної роботи	Бали
1. Аудиторна 1.1. Лекція	<ul style="list-style-type: none"> <li>Відвідування всіх лекцій лекції та лабораторних</li> </ul>	20
1.2. Практичні заняття	<ul style="list-style-type: none"> <li>Підготовка до лабораторної роботи та її виконання (4x10=40)</li> </ul>	40

Форма навчальної роботи	Вид навчальної роботи	Бали
2. Підсумковий контроль.	МКР№1	20
	МКР№2	20
Усього за семестр		100

2 семестр

Форма навчальної роботи	Вид навчальної роботи	Бали
1. Аудиторна 1.1. Лекція	<ul style="list-style-type: none"> <li>Відвідування всіх лекцій лекції та лабораторних</li> </ul>	20
1.2. Практичні заняття	<ul style="list-style-type: none"> <li>Підготовка до лабораторної роботи та її виконання (3x10=40)</li> </ul>	30
2. Підсумковий контроль.	МКР№1	5
	МКР№2	5
	Екзамен	40
Усього за семестр		100

Додаткові бали:

Форма роботи	Вид роботи	Бали
1. Навчальна	1. Участь в предметних олімпіадах: університетських, міжвузівських, всеукраїнських, міжнародних	5
	2. Участь в конкурсах на кращого знавця дисципліни: університетських, міжвузівських, всеукраїнських, міжнародних	3
2. Науково-дослідна	1. Участь в наукових гуртках	2
	2. Участь в наукових студентських клубах	2
	3. Участь в наукових магістерських семінарах	2
	4. Участь в конкурсах студентських робіт: університетських, міжвузівських, всеукраїнських, міжнародних	5
	5. Участь в наукових студентських конференціях: університетських, міжвузівських, всеукраїнських, міжнародних	5

**Шкала оцінювання здобувачів вищої освіти за результатами вивчення навчальної дисципліни**

<i>Сума балів за всі види навчальної діяльності</i>	<i>Оцінка за шкалою ЄКТС</i>	<i>Оцінка за національною шкалою</i>
90-100	A	Відмінно
82-89	B	Дуже добре
74-81	C	Добре
64-73	D	Задовільно
60-63	E	Задовільно достатньо
35-59	FX	Незадовільно з можливістю повторного складання
0-34	F	Незадовільно з обов'язковим повторним вивченням навчальної дисципліни

## **Розділ 6. Інформаційні джерела**

### **Основна**

1. Файн Я. Програмування на Java для дітей, батьків, дідусів та бабусь / Я. Файн — 2014.
2. Васильєв О. М. Програмування мовою Java / О. М. Васильєв - Bohdan Books - 2022.
3. Schildt, Herbert. Java: a beginner's guide. McGraw-Hill Education, 2022.
4. Downey, Allen B., and Chris Mayfield. Think Java: How to think like a computer scientist. O'Reilly Media, 2019.

### **Додаткова**

5. Oaks, Scott. Java performance: in-depth advice for tuning and programming Java 8, 11, and beyond. " O'Reilly Media, Inc.", 2020.
6. Farrell, Joyce. Java programming. Cengage Learning, 2022.
7. Sedgewick, Robert, and Kevin Wayne. Introduction to programming in Java: an interdisciplinary approach. Addison-Wesley Professional, 2017.
8. N. Singh, S. S. Chouhan and K. Verma, "Object Oriented Programming: Concepts, Limitations and Application Trends,"*2021 5th International Conference on Information Systems and Computer Networks (ISCON)*, 2021, pp. 1-4, doi: 10.1109/ISCON52037.2021.9702463.
9. Lartey W. H. Algorithmization and development of simulator on the topic "Cycles in java" of the discipline "Programming" / W. H. Lartey, Yu. F. Oleksiichuk // Комп'ютерні науки і прикладна математика (КНіПМ-2021): матеріали наук.-практ. семінару. Випуск 6. / За ред. Ємця О.О. – Полтава: Кафедра ММСІ ПУЕТ, 2021
10. Олексійчук Ю. Ф. Програмування: навчально-методичний посібник для самостійного вивчення дисципліни студентами напряму підготовки 6.040302 «Інформатика» ПУЕТ / Ю. Ф. Олексійчук. – Полтава: ПУЕТ, 2015. – 31 с.

## **Розділ 7. Програмне забезпечення навчальної дисципліни**

Використовується наступне програмне забезпечення:

1. Середовище розробки для мови програмування Java NetBeans (або Eclipse, або IntelliJ IDEA).