

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
ІМЕНІ ТАРАСА ШЕВЧЕНКА**

**ФАКУЛЬТЕТ КОМП'ЮТЕРНИХ НАУК ТА КІБЕРНЕТИКИ
КАФЕДРА ІНТЕЛЕКТУАЛЬНИХ ПРОГРАМНИХ СИСТЕМ**

«ЗАТВЕРДЖУЮ»

Заступник декана
з навчальної роботи

_____ Кашпур О.Ф.

«___» _____ 2019 року

**РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ
для студентів**

галузь знань **12 Інформаційні технології**
спеціальність **121 Інженерія програмного забезпечення**
освітній рівень **бакалавр**
освітня програма **Програмна інженерія**
вид дисципліни **обов'язкова**

Форма навчання	денна
Навчальний рік	2021/2022
Семестр	5,6
Кількість кредитів ECTS	8
з них семестр 5	3
семестр 6	4
Мова викладання, навчання та оцінювання	українська
Форма заключного контролю	іспит

Викладачі: **к.ф.-м.н., доц. Галкін О.В.** (лекції, лабораторні заняття)
к.т.н., доц. Демківський Є.О. (лабораторні заняття)

Пролонговано: на 20__/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

на 20__/20__ н.р. _____ (_____) «__» __ 20__ р.
(підпис, ПІБ, дата)

КИЇВ 2019

Розробник: Галкін Олександр Володимирович, к.ф.-м.н., доц.,
доцент кафедри інтелектуальних програмних систем

ЗАТВЕРДЖЕНО

Завідувач кафедри інтелектуальних програмних систем

_____ О.І. Провотар

Протокол № __ від «__» _____ 2019 р.

Схвалено науково-методичною комісією факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2019 року №__

Голова науково-методичної комісії _____ Л.Л. Омельчук

«_____» _____ 2019 року

Затверджено вченою радою факультету комп'ютерних наук та кібернетики

Протокол від «__» _____ 2019 року №__

Голова вченої ради факультету _____ А.В. Анісімов

1. Мета дисципліни – опанування об'єктно-орієнтованої парадигми програмування на прикладі мови програмування Java, включаючи розгляд особливостей Java-технологій.

2. Попередні вимоги до опанування навчальної дисципліни

1. **Знати:** базові поняття математичного аналізу, лінійної та загальної алгебри, диференціальних рівнянь, програмування, концепції об'єктно-орієнтованого програмування.

2. **Вміти:** програмувати на мові C++.

3. Анотація навчальної дисципліни

Навчальна дисципліна «Об'єктно-орієнтоване програмування» є складовою освітньо-професійної програми підготовки фахівців за першим (бакалаврським) рівнем вищої освіти галузі знань 12 Інформаційні технології за спеціальністю 121 Інженерія програмного забезпечення освітньо-професійної програми „Програмна інженерія”.

Дисципліна належить до блоку обов'язкових навчальних дисциплін. Викладається у **5 та 6 семестрах 3 курсу в обсязі – 210 год. (7 кредитів ECTS)**, зокрема: лекції – 56 год., лабораторні заняття – 42 год., консультації – 4 год., самостійна робота – 108 год. У курсі передбачено 4 частини та 4 контрольні роботи, **залік в 5 семестрі та іспит в 6 семестрі**.

В результаті вивчення навчальної дисципліни студент повинен:

знати основні парадигми програмування; технології програмування на Java;

вміти створювати програми на мові програмування Java; володіти прийомами об'єктно-орієнтованого програмування, знати основні принципи спадкування; будувати основні діаграми UML і розуміти їх призначення; створювати web-орієнтовані системи на мові Java.

Дисципліна є логічним продовженням, доповненням та розширенням дисциплін «Основи об'єктно-орієнтованого програмування», «Розробка WEB-орієнтованих систем», «Організація баз даних та знань» та виступає базовою для обов'язкових дисциплін «Програмна інженерія» і «Основи побудови компіляторів» та дисципліни «Програмування мовою Python» блоку вибору «Інтелектуальні системи» освітньої програми «Програмна інженерія».

4. Завдання (навчальні цілі). Основними завданнями дисципліни «Об'єктно-орієнтоване програмування» є набуття знань, умінь та навичок (компетенцій) на рівні новітніх досягнень в області об'єктно-орієнтованого програмування відповідно до освітньої кваліфікації бакалавр з програмної інженерії.

Зокрема, розвивати:

- Здатність до абстрактного мислення, аналізу та синтезу (ЗК01).
- Здатність застосовувати знання у практичних ситуаціях (ЗК02).
- Здатність спілкуватися державною мовою як усно, так і письмово (ЗК03).
- Здатність вчитися і оволодівати сучасними знаннями (ЗК05).
- Здатність до пошуку, оброблення та аналізу інформації з різних джерел (ЗК06).
- Здатність працювати в команді (ЗК07).
- Здатність аналізувати предметні області, ідентифікувати, класифікувати та формулювати вимоги (СК01).
- Здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування (СК02).
- Здатність розробляти архітектури, модулі та компоненти програмних систем (СК03).
- Здатність дотримуватися специфікацій, стандартів, правил і рекомендацій в професійній галузі при реалізації процесів життєвого циклу (СК05).

- Володіння знаннями про інформаційні моделі даних та системи, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних (СК07).
- Здатність реалізовувати фази та ітерації життєвого циклу програмних систем та інформаційних технологій на основі відповідних моделей і підходів розробки програмного забезпечення (СК11).
- Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення (СК13).
- Здатність до алгоритмічного та логічного мислення (СК14).

5. Результати навчання за дисципліною.

Результат навчання (1. знати; 2. вміти; 3. комунікація; 4. автономність та відповідальність)		Форми (та/або методи і технології) викладання і навчання	Методи оцінювання та пороговий критерій оцінювання (за необхідності)	Відсоток у підсумковій оцінці з дисципліни
Код	Результат навчання			
РН1.1	Знати основні парадигми програмування	Лекція	Контрольна робота, іспит	5%
РН1.2	Знати технології програмування на Java	Лекція, лабораторне заняття, самостійна робота	Контрольна робота, іспит	25%
РН2.1	Вміти створювати програми на мові програмування Java	Лабораторне заняття, самостійна робота	Контрольна робота, захист лабораторної роботи, іспит	25%
РН2.2	Вміти володіти прийомами об'єктно-орієнтованого програмування, знати основні принципи спадкування	Лабораторне заняття, самостійна робота	Контрольна робота, захист лабораторної роботи, іспит	10%
РН2.3	Вміти будувати основні діаграми UML і розуміти їх призначення	Лабораторне заняття, самостійна робота	Контрольна робота, захист лабораторної роботи, іспит	10%
РН2.4	Вміти створювати web-орієнтовані системи на мові Java	Лабораторне заняття, самостійна робота	Контрольна робота, захист лабораторної роботи, іспит	10%
РН3.1	Обґрунтовувати власний погляд на розв'язання задачі, спілкуватися з колегами з питань проектування та розробки програм, складати письмові звіти	Лабораторне заняття, самостійна робота	Поточне оцінювання, захист лабораторної роботи	5%
РН4.1	Організувати свою самостійну роботу для досягнення результату	Самостійна робота	Поточне оцінювання, захист лабораторної роботи	5%
РН4.2	Відповідально ставитися до виконуваних робіт, нести відповідальність за їх якість	Лабораторне заняття	Захист лабораторної роботи	5%

6. Співвідношення результатів навчання дисципліни із програмними результатами навчання.

Програмні результати навчання	Результати навчання дисципліни								
	РН1.1	РН1.2	РН2.1	РН2.2	РН2.3	РН2.4	РН3.1	РН4.1	РН4.2
ПРН01. Аналізувати, цілеспрямовано шукати і вибирати необхідні для вирішення професійних завдань інформаційно-довідникові ресурси і знання з урахуванням сучасних досягнень науки і техніки.	+	+	+	+				+	
ПРН03. Знати основні процеси, фази та ітерації життєвого циклу програмного забезпечення.		+							
ПРН04. Знати і застосовувати професійні стандарти і інші нормативно-правові документи в галузі інженерії програмного забезпечення.		+	+	+	+				
ПРН05. Знати і застосовувати відповідні математичні поняття, методи доменного, системного і об'єктно-орієнтованого аналізу та математичного моделювання для розробки програмного забезпечення.	+	+	+	+	+	+		+	+
ПРН06. Уміння вибирати та використовувати відповідну задачі методологію створення програмного забезпечення.			+	+					
ПРН07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.	+	+	+	+	+	+			
ПРН12. Застосовувати на практиці ефективні підходи щодо проектування програмного забезпечення.				+	+				
ПРН13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.	+	+	+	+	+				
ПРН14. Застосовувати на практиці інструментальні програмні засоби доменного аналізу, проектування, тестування, візуалізації, вимірювань та документування програмного забезпечення.					+				
ПРН15. Мотивовано обирати мови програмування та технології розробки для розв'язання завдань створення і супроводження програмного забезпечення.	+	+					+		
ПРН16. Мати навички командної розробки, погодження, оформлення і випуску всіх видів програмної документації.					+		+	+	+
ПРН17. Вміти застосовувати методи компонентної розробки програмного забезпечення.			+	+	+	+			
ПРН18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.		+	+			+			
ПРН19. Знати та вміти застосовувати методи верифікації та валідації програмного забезпечення.					+				

7. Схема формування оцінки.

7.1. Форми оцінювання студентів.

– семестрове оцінювання (максимальна кількість балів):

Семестр 5

1. Контрольна робота 1: РН 1.1, РН 1.2, РН 2.1, РН 2.2 – **20/12 б.**
2. Контрольна робота 2: РН 1.1, РН 1.2, РН 2.1–2.3 – **20/12 б.**

3. Лабораторні роботи (3): РН 1.1-1.2, РН 2.1-2.3, РН 3.1, РН 4.1-4.2 – **20/12 б. кожна**

Підсумкове оцінювання (у формі заліку):

- Залікові бали визначаються як сума оцінок/балів за всіма успішно оціненими результатами навчання передбачених даною програмою.
- Оцінки нижче від мінімального порогового рівня не додаються.
- Мінімальний пороговий рівень для сумарної оцінки за всіма компонентами становить 60% від максимально можливої кількості балів.

Семестр 6

1. Контрольна робота 1: РН 1.1, РН 1.2, РН 2.1, РН 2.2 – **7/4 б.**

2. Контрольна робота 2: РН 1.1, РН 1.2, РН 2.1–2.2, РН 2.4 – **8/5 б.**

3. Лабораторні роботи (3): РН 1.1-1.2, РН 2.1-2.4, РН 3.1, РН 4.1-4.2 – **15/9 б. кожна (разом 45/27 б.)**

Підсумкове оцінювання (у формі іспиту):

- максимальна кількість балів які можуть бути отримані студентом: 40 балів;
- результати навчання які будуть оцінюватись: РН 1.1, РН 1.2, РН 2.1–2.4;
- форма проведення і види завдань: письмово-усна форма, 4 практичні завдання по 10 балів (25%) кожне.

Студент **допускається** до іспиту, якщо він під час семестру набрав **більше 36 балів**, причому лабораторні роботи було виконано **як мінімум на 60%**.

Для отримання загальної позитивної оцінки з дисципліни оцінка за іспит **не може бути меншою 24 балів**.

7.2. Організація оцінювання.

Терміни проведення форм оцінювання в 5 семестрі:

1. Контрольна робота 1: до 6 тижня семестру.
2. Контрольна робота 2: до 14 тижня семестру.
3. Лабораторна робота 1 (проект): до 4 тижня семестру (лабораторне заняття №2).
4. Лабораторна робота 2 (проект): до 8 тижня семестру (лабораторне заняття №4).
5. Лабораторна робота 3 (проект): до 14 тижня семестру (лабораторне заняття №7).

Терміни проведення форм оцінювання в 6 семестрі:

1. Контрольна робота 1: до 6 тижня семестру.
2. Контрольна робота 2: до 14 тижня семестру.
3. Лабораторна робота 1 (проект): до 4 тижня семестру (лабораторне заняття №4).
4. Лабораторна робота 2 (проект): до 9 тижня семестру (лабораторне заняття №9).
5. Лабораторна робота 3 (проект): до 14 тижня семестру (лабораторне заняття №14).

Студент має право один раз перескласти контрольну роботу з можливістю отримати не більше 80% балів, призначених за роботу. Термін перескладання визначається викладачем.

У випадку відсутності студента з поважних причин відпрацювання та перездачі контрольних робіт здійснюються у відповідності до „Положення про порядок оцінювання знань студентів при кредитно-модульній системі організації навчального процесу” від 1 жовтня 2010 року.

У разі неякісного або невчасного виконання лабораторної роботи викладач має право не зарахувати завдання або знизити за нього бали.

Студент має право здавати лабораторні роботи після закінчення визначеного для них терміну, але з втратою 10% балів за кожен тиждень, що пройшов від закінчення терміну її здачі.

7.3. Шкала відповідності оцінок

Відмінно / Excellent	90-100
Добре / Good	75-89
Задовільно / Satisfactory	60-74
Незадовільно / Fail	0-59
Зараховано / Passed	60-100
Не зараховано / Fail	0-59

8. Структура навчальної дисципліни. Тематичний план лекцій.

СЕМЕСТР 5

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабор. заняття	С/р
Частина 1. Опис Java Runtime Environment				
1	Тема 1. Програмне середовище Java, компілятор, компоувальник. Вхідна, об'єктна і виконавча програми. Роздільна компіляція. Програмний модуль	2		3
2	Тема 2. Класи та об'єкти. Поля в класах. Методи класів. Конструктори класів. Ініціалізація параметрів в класах. Посилання this	2	2	3
3–4	Тема 3. Перевантаження методів. Статичні члени класів. Статичні методи класів. Збірка сміття. Метод finalize. Метод main. Спадкування в Java	4	2	6
5–6	Тема 4. Конструктори в нащадків. Перевизначення методів. Ключове слово super. Методи та класи з модифікатором final. Клас Object. Абстрактні класи. Інтерфейс cloneable	4	2	6
Контрольна робота №1				2
		12	6	20
Частина 2. Виключення та багатопоточність в Java				
7	Тема 5. Поняття інтерфейсу. Розширення інтерфейсів. Реалізація інтерфейсів. Використання реалізації інтерфейсів. Коментарі в Java. Змінні. Основні типи в Java. Перетворення типів. Масиви. Явне приведення типів	2	2	3
8–9	Тема 6. Виключення в Java. Оператор throw. Умова throws. Оператори try, catch, finally. Рядки	4	2	5

10–11	Тема 7. Потоків в Java. Створення потоків. Синхронізація. Оператор synchronized. Методи wait, notify, notifyall. Планування потоків. Призупинення потоків. Завершення роботи потоків	4	2	6
12–13	Тема 8. Введення та вивід у Java. Клас InputStream. Клас OutputStream. Стандартні типи потоків. Фільтруючі потоки. Клас PrintStream. Байтові потоки. Конвейерні потоки	4	2	6
14	Тема 9. Вступ до UML	2		4
Контрольна робота №2				2
		16	8	26
ВСЬОГО		28	14	46

Загальний обсяг – **90 год.**, у тому числі:

Лекцій – **28 год.**,

Консультацій – **2 год.**,

Лабораторних занять – **14 год.**,

Самостійна робота – **46 год.**

Тематика лабораторних проєктів

1. Робота з OpenGL.
2. Об'єктно-орієнтоване моделювання.
3. Робота з XML.

Перелік питань для підготовки до заліку

4. Поняття класу в Java.
5. Поняття об'єкта.
6. Поняття методу та поля класу.
7. Поняття компілятора.
8. Поняття компонування.
9. Константи та змінні.
10. Ініціалізація змінних.
11. Розширення інтерфейсів.
12. Перетворення типів при перевантаженні методів.
13. Явне приведення типів.
14. Виключення в Java.
15. Оператор throw.
16. Створення потоків.
17. Оператор synchronized.
18. Планування потоків.
19. Завершення роботи потоків.
20. Введення та вивід в Java.
21. Поняття про UML.

СЕМЕСТР 6

№ лекції	Назва лекції	Кількість годин		
		Лекції	Лабор. заняття	С/р
Частина 3. Рефлексія, серіалізація та класичні колекції в Java				

15	Тема 10. Класи-оболонки. Рефлексія. Клас Class	2	2	6
16–17	Тема 11. Серіалізація об'єктів. Байтові потоки Object. Підготовка класів до серіалізації. Порядок серіалізації та десеріалізації. Налаштування механізму серіалізації. Типи PutField та GetField. Інтерфейс Externalizable	4	4	8
18–19	Тема 12. Колекції. Інтерфейс iterator, Comparable та Comparator. Інтерфейс Set, SortedSet, HashSet, TreeSet, List. Клас ArrayList	4	4	8
20	Тема 13. Інтерфейс користувача. Компоненти JavaAWT. Компоненти JavaSwing. Допоміжні компоненти. Графічні та мультимедіа компоненти	2	2	4
Контрольна робота №3				2
		12	12	28
Частина 4. Основи web-програмування в Java				
21–22	Тема 14. Сервлети. Архітектура сервлетів. Інтерфейси сервлетів. http інтерфейси сервлетів. Обробка запиту. Генерація відповідей. Керування сеансом. JSP	4	4	8
23–24	Тема 15. XML. Формат XML-документа. DTD опис. Гіпертекстові засоби XML. Технологія RMI	4	4	8
25–26	Тема 16. Побудова компонентів за допомогою Enterprise JavaBeans. Конфігурація та налаштування EJB. Компоненти сеансу на боці сервера. Клієнтські інтерфейси. Компоненти об'єктів на боці сервера	4	4	8
27–28	Тема 17. Організація доступу до даних. JDBC. Робота з базами даних з використанням Hibernate	4	4	8
Контрольна робота №4				2
		16	16	34
ВСЬОГО		28	28	62

Загальний обсяг– **120 год.**, у тому числі:

Лекцій – **28 год.**,

Консультацій – **2 год.**,

Лабораторних занять – **28 год.**,

Самостійна робота – **62 год.**

Тематика лабораторних проектів

1. Створення клієнт-серверного-додатку.
2. Створення web-орієнтованої системи за допомогою сервлетів та технології JDBC.
3. Створення web-орієнтованої системи за допомогою Spring-фреймворку.

Перелік питань для підготовки до іспиту

1. Серіалізація об'єктів.
2. Підготовка класів до серіалізації.
3. Налаштування механізму серіалізації.
4. Рефлексія в Java.
5. Поняття сервлета.
6. Протокол http.
7. Організація доступу до даних в Java.

8. Побудова компонентів за допомогою Enterprise JavaBeans.
9. Поняття про JSP.
10. Передача об'єктів по значенню та RMI.
11. Поняття XML
12. DTD опис.
13. Гіпертекстові засоби XML.

9. Рекомендовані джерела.

1. О.В. Галкін, Л.О. Катеринич, О.С. Шкільняк. Програмування на JAVA 8: навчальний посібник. – К.: Логос, 2017. – 185 с.
2. К. Арнольд, Дж. Гослинг, Д. Холмс. Язык программирования Java, 3-е изд. – К.: Вильямс, 2001. – 624 с.
3. Б. Эккель. Философия Java, 4-е изд. – СПб: Питер, 2016. – 1168 с.
4. Д. Блох. Java. Эффективное программирование, 2-е изд. – М.: Лори, 2014. – 440 с.
5. С. Макконнелл. Совершенный код, 2-е изд. – Русская редакция, 2017. – 896 с.
6. К.С. Хорстманн, Г. Корнелл. Библиотека профессионала. Java 2. Том 1. Основы, 9-изд. – М.: Вильямс, 2015. – 864 с.
7. К.С. Хорстманн, Г. Корнелл. Библиотека профессионала. Java 2. Том 2. Тонкости программирования, 9-е изд. – М.: Вильямс, 2013. – 1008 с.
8. И.Н. Блинов, В.С. Романчик. Java. Методы программирования. – Минск: “Четыре четверти”, 2013. – 896 с.
9. П. Ноутон, Г. Шилдт. Java 2. Наиболее полное руководство. – СПб.: БХВ-Петербург, 2008. – 1072 с.
10. Б.У. Перри. Java сервлеты и JSP. Сборник рецептов. – М: Кудиц-пресс, 2009. – 768 с.
11. М. Холл. Сервлеты и JavaServer Pages. – СПб.: Питер, 2002. – 496 с.