

МІЖНАРОДНИЙ ЄВРОПЕЙСЬКИЙ УНІВЕРСИТЕТ
Навчально-науковий інститут «Європейська школа бізнесу»
Кафедра інформаційних технологій

ЗАТВЕРДЖЕНО

Директор ННІ

«Європейська школа бізнесу»



Юлія РЕМИГА

від « 11 » 09 2023 р.

М.П.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

Рівень вищої освіти перший (бакалаврський)

Спеціальність 121 «Інженерія програмного забезпечення»

Освітня програма «Інженерія програмного забезпечення»

Робоча програма навчальної дисципліни «Об'єктно-орієнтоване програмування» складена на основі освітньо-професійної програми 121 «Інженерія програмного забезпечення» для першого (бакалаврського) рівня спеціальності 121 «Інженерія програмного забезпечення», затвердженої Вченою радою Університету «30» травня 2023 року, протокол № 4.

Укладач: Нестеренко Олександр Васильович, доктор технічних наук, доцент

Рецензент: Фаловський Олександр Олександрович, к.т.н.

Гарант освітньої програми:  Олександр НЕСТЕРЕНКО, доктор технічних наук, доцент

Робочу програму навчальної дисципліни розглянуто та схвалено кафедрою інформаційних технологій, протокол від 31.08.2023 р. № 1.

ВСТУП

Програма вивчення навчальної дисципліни «Об’єктно-орієнтоване програмування» складена відповідно до Стандарту вищої освіти України (далі – Стандарт) галузі знань 12 «Інформаційні технології» спеціальності 121 «Інженерія програмного забезпечення».

Опис навчальної дисципліни (анотація). Дана навчальна дисципліна належить до обов’язкових компонентів освітньої програми «Інженерія програмного забезпечення» підготовки майбутніх розробників програмного забезпечення.

Таблиця 1

Найменування показників	Галузь знань, напрям підготовки, освітній рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 8	Галузь знань, 12 «ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ»	Нормативна	
Розділів – 2	Спеціальність: 121 «ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»	Рік підготовки	
Змістових розділів – 4		2	2
Індивідуальне науково-дослідне завдання: стартап		Семестр	
		3,4	3,4
		Лекції	
		32	10
		Лабораторні	
	64	10	
Тижневе навантаження: аудиторних – 3	Освітній рівень: бакалавр	Самостійна робота	
самостійної роботи студента – 4		144	220
		Вид контролю:	
	залік, екзамен	залік, екзамен	

Предметом вивчення навчальної дисципліни є синтаксис, основні елементи, способи та прийоми об’єктно-орієнтованого програмування мовами Java, C++ та C#.

Міждисциплінарні зв’язки: програма упорядкована відповідно до анотації освітньо-професійної програми підготовки бакалаврів, базується на вивченні дисциплін «Методи та засоби інформаційних технологій», «Основи програмування», передуює вивченню нормативних дисциплін «Архітектура та проектування програмного забезпечення», «Конструювання програмного забезпечення», «Емпіричні методи програмної інженерії».

Знання, отримані здобувачами вищої освіти під час вивчення дисципліни «Об’єктно-орієнтоване програмування» можуть бути застосовані під час проходження виробничої практики, підготовки курсових робіт та кваліфікаційних робіт за спеціальністю.

1. МЕТА ТА ЗАВДАННЯ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

1.1. **Метою** вивчення дисципліни «Об'єктно-орієнтоване програмування» є опанування здобувачами вищої освіти принципів створення програмного забезпечення на основі об'єктно-орієнтованої парадигми, знань щодо сучасних стандартів, підходів, методів та засобів об'єктно-орієнтованого програмування, оволодіння компетенціями ефективного використання ООП для створення алгоритмічно та технологічно навантажених програмних проектів. Знайомство з підходами до розробки індустріального та академічного програмного забезпечення

1.2. Основними **завданнями** вивчення дисципліни «Об'єктно-орієнтоване програмування» є:

- вивчити основи проектування програмного забезпечення на сучасній технологічній основі об'єктно-орієнтованого програмування;
- вивчити прийоми роботи з візуальними середовищами програмування;
- формування у студентів навичок абстрактного мислення, що повинне допомогти вирішенню прикладних задач, пов'язаних з різноманітними галузями діяльності.

1.3. **Компетентності та результати навчання**, формуванню яких сприяє дисципліна (взаємозв'язок з нормативним змістом підготовки здобувачів вищої освіти, сформульованим у термінах результатів навчання у Стандарті).

Згідно з вимогами стандарту дисципліна забезпечує набуття студентами **компетентностей** (Таблиця 2):

Таблиця 2

Інтегральна компетентність	Здатність розв'язувати складні спеціалізовані завдання або практичні проблеми інженерії програмного забезпечення, що характеризуються комплексністю та невизначеністю умов, із застосуванням теорій та методів інформаційних технологій.
Загальні компетентності	ЗК 01. Здатність до абстрактного мислення, аналізу та синтезу.
Спеціальні (фахові, предметні) компетентності	СК7. Володіння знаннями про інформаційні моделі даних, здатність створювати програмне забезпечення для зберігання, видобування та опрацювання даних. СК13. Здатність обґрунтовано обирати та освоювати інструментарій з розробки та супроводження програмного забезпечення.

Інтегративні кінцеві програмні результати навчання, формуванню яких сприяє навчальна дисципліна:

Програмні результати навчання	<p>ПР07. Знати і застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення.</p> <p>ПР13. Знати і застосовувати методи розробки алгоритмів, конструювання програмного забезпечення та структур даних і знань.</p> <p>ПР18. Знати та вміти застосовувати інформаційні технології обробки, зберігання та передачі даних.</p>
--------------------------------------	--

Після опанування дисципліни студент повинен

знати:

- основні програмні середовища і засоби розробки програм на мовах Java, C++ та C#;
- відмінності між об'єктно-орієнтованим і функціональним програмуванням;
- принципи побудови класів об'єктів та їх методів, основні типи класів та способи їх надбудови та взаємодії;
- правила побудови і основні елементи об'єктно-орієнтованих програм різними мовами;
- основні принципи ООП;
- принципи організації структур збереження даних і методи доступу до даних;
- порядок створення класів та їх будову;
- організацію безпечного доступу до інформації завдяки застосуванню засобів інкапсуляції;
- основні прийоми роботи з одно-, двовимірними статичними і динамічними масивами;
- методи повторного використання коду та інформаційних компонентів через механізми спадкування класів та поліморфізму;
- методи створення і використання технології виключень для підвищення надійності програмного забезпечення;
- засоби стандартних бібліотек для реалізації інтерфейсів взаємодії з людиною;

уміти:

- використовувати Інтернет для пошуку документації, специфікацій, типових способів вирішення завдань при розробці програм;
- будувати та зображати різними способами функціональні та структурні схеми програмного забезпечення,
- оперувати моделями об'єктів (даними й методами) та створювати програмні засоби керування ними;
- користуватися середовищем розробки IntelliJ IDEA і мовою Java, володіти технологією об'єктно-орієнтованого програмування;
- обґрунтовувати та проектувати ієрархію класів та об'єктів проекту;

- застосовувати при розробці класів інкапсуляцію, поліморфізм та наслідування (в т.ч. множинне);
- створювати і обробляти файли та інформацію, яка зберігається в них;
- здійснювати управління програмними потоками;
- використовувати різні технології програмування на мовах Java, C++ та C #;
- створювати віконне та консольне ПЗ;
- виконувати тестування та налагодження програми.

2. ІНФОРМАЦІЙНИЙ ОБСЯГ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

РОЗДІЛ 1

ЗМІСТОВИЙ РОЗДІЛ 1

ЗАГАЛЬНІ УЯВЛЕННЯ ПРО МОВУ JAVA

Тема 1.1. Основи програмування на Java

Тема 1.2. Робота з рядками.

Тема 1.3. Лямбда-вирази.

ЗМІСТОВИЙ РОЗДІЛ 2

КОНЦЕПЦІЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ

Тема 2.1. Основні поняття об'єктно-орієнтованого програмування.

Тема 2.2. Структури даних. Колекції

Тема 2.3. Додаткові класи. Обробка виключень. Регулярні вирази

Тема 2.4. Потоки введення-виведення. Робота з файлами

Тема 2.5. Багатопоточне програмування.

ЗМІСТОВНИЙ РОЗДІЛ 3

СТВОРЕННЯ ЗАСТОСУВАНЬ НА JAVA

Тема 3.1. Створення графічного інтерфейсу користувача на Java.

Тема 3.2. Аплети в Java.

Тема 3.3. Розробка web-аплікацій на Java.

Тема 3.4. Android-програмування.

РОЗДІЛ 2

ЗМІСТОВНИЙ РОЗДІЛ 4

ЗАСОБИ РЕАЛІЗАЦІЇ ООП В МОВІ C++

Тема 4.1. Аналіз засобів реалізації ООП в мові C++.

Тема 4.2. Клас та об'єкт в C++.

Тема 4.3. Успадкування як ключовий момент ООП.

Тема 4.4. Реалізація поліморфізму в C++.

ЗМІСТОВНИЙ РОЗДІЛ 5

ЗАСОБИ РЕАЛІЗАЦІЇ ООП В МОВІ C# НА ПЛАТФОРМІ .NET

Тема 5.1. Знайомство с C#.NET.

Тема 5.2. Основи реалізації ООП в C#.

Тема 5.3. Технології інтеграції запитів в мові C# (LINQ).

Тема 5.4. Розробка веб-додатків з застосуванням C#.Net.

Назви розділів і тем	Кількість годин			
	Всього	у тому числі		
		Лекції	Лабораторні	СРС
Розділ 1.				
Змістовний розділ 1. ЗАГАЛЬНІ УЯВЛЕННЯ ПРО МОВУ JAVA				
<i>Тема 1.1.</i> Основи програмування на Java	8	2		6
<i>Тема 1.2.</i> Робота з рядками	11	1	4	6
<i>Тема 1.3.</i> Лямбда-вирази	11	1	4	6
<i>Разом за змістовним розділом 1</i>	30	4	8	18
Змістовний розділ 2. КОНЦЕПЦІЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ				
<i>Тема 2.1.</i> Основні поняття об'єктно-орієнтованого програмування	8	2		6
<i>Тема 2.2.</i> Структури даних. Колекції	9	1	2	6
<i>Тема 2.3.</i> Додаткові класи. Обробка виключень. Регулярні вирази	9	1	2	6
<i>Тема 2.4.</i> Потоки введення-виведення. Робота з файлами	11	1	4	6
<i>Тема 2.5.</i> Багатопоточне програмування	9	1	2	6
<i>Разом за змістовним розділом 2</i>	46	6	10	30
Змістовний розділ 3. СТВОРЕННЯ ЗАСТОСУВАНЬ НА JAVA				
<i>Тема 3.1.</i> Створення графічного інтерфейсу користувача на Java	10	2	2	6
<i>Тема 3.2.</i> Аплети в Java	10	2	4	6
<i>Тема 3.3.</i> Розробка web-аплікацій на Java	11	1	4	6
<i>Тема 3.4.</i> Android-програмування	11	1	4	6
<i>Разом за змістовним розділом 3</i>	42	6	14	24
<i>Разом за розділом 1</i>	120	16	32	72
Розділ 2.				
Змістовний розділ 4. ЗАСОБИ РЕАЛІЗАЦІЇ ООП В МОВІ C++				
<i>Тема 4.1.</i> Аналіз засобів реалізації ООП в мові C++	15	2	4	9
<i>Тема 4.2.</i> Клас та об'єкт в C++	15	2	4	9
<i>Тема 4.3.</i> Успадкування як ключовий момент ООП	15	2	4	9
<i>Тема 4.4.</i> Реалізація поліморфізму в C++	15	2	4	9
<i>Разом за змістовним розділом 4</i>	60	8	16	36
Змістовний розділ 5. ЗАСОБИ РЕАЛІЗАЦІЇ ООП В МОВІ C++				
<i>Тема 5.1.</i> Знайомство с C#.NET	15	2	4	9
<i>Тема 5.2</i> Основи реалізації ООП в C#	15	2	4	9

Тема 5.3. Технології інтеграції запитів в мові С# (LINQ)	15	2	4	9
Тема 5.4. Розробка веб-додатків з застосуванням С#.Net	15	2	4	9
Разом за змістовним розділом 5	60	8	16	36
Разом за розділом 2	120	16	32	72
Всього	240	32	64	144

4. ТЕМИ ЛЕКЦІЙ

№ лекції	Назва теми лекції та перелік основних питань
1	Тема 1.1. ОСНОВИ ПРОГРАМУВАННЯ НА JAVA. Що таке Java. Віртуальна Java-машина. Байт-код. JIT-компіляція. Категорії програм, написаних на мові Java. Середовища розробки Java-проектів. Структура програми. Змінні і константи. Типи даних. Консольне введення / виведення в Java. Арифметичні операції. Побітові операції. Умовні вирази. Операції присвоювання і пріоритет операцій. Перетворення базових типів даних. Умовні конструкції. Цикли. Масиви. Методи. Параметри методів. Оператор return. Перевантаження методів. Рекурсивні функції. Обробка виняткових ситуацій.
2	Тема 1.2. РОБОТА З РЯДКАМИ. Об'єкти (змінні) типу String. Перевірка двох рядків на рівність. Довжина рядка. Визначення символу за заданим індексом. Об'єднання рядків. Визначення наявності підрядка в рядку. Розбиття рядка на масив цілих чисел. Поняття масиву рядків. Загальна форма оголошення одновимірного масиву рядків. Двовимірний масив рядків. Довжина масиву рядків. Пошук заданого рядка в одновимірному масиві рядків. Сортування одновимірного масиву рядків за алфавітом методом вставки. Визначення кількості входжень заданого рядка у двовимірному масиві рядків. Заміна рядка у двовимірному масиві рядків. Тема 1.3. ЛЯМБДА-ВИРАЗИ. Поняття лямбда-виразів. Лямбда як параметри і результати методів. Вбудовані функціональні інтерфейси.
3	Тема 2.1. ОСНОВНІ ПОНЯТТЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ. Принципи ООП. Класи і об'єкти. Пакети. Модифікатори доступу і інкапсуляція. Статичні члени і модифікатор static. Об'єкти як параметри методів. Внутрішні і вкладені класи. Спадкування. Абстрактні класи. Ієрархія. наслідування і перетворення типів. Інтерфейси. Інтерфейси в механізмі зворотного виклику. Перерахування enum. Клас Object і його методи. Узагальнення (Generics). Обмеження узагальнень. Спадкування і узагальнення. Довідкові типи і копіювання об'єктів.
4	Тема 2.2. СТРУКТУРИ ДАНИХ. КОЛЕКЦІЇ. Інтерфейс Enumeration. Клас BitSet. Клас Vector. Клас Stack. Клас Dictionary. Клас Hashtable (хеш-таблиця). Клас Properties. Типи колекцій. інтерфейс Collection. Клас ArrayList і інтерфейс List. Черги і клас ArrayDeque. клас LinkedList. Інтерфейс Set і клас HashSet. SortedSet, NavigableSet, TreeSet. Інтерфейси Comparable і Comparator. Сортування. Тема 2.3. ДОДАТКОВІ КЛАСИ. ОБРОБКА ВИКЛЮЧЕНЬ. Математичні обчислення і клас Math. Великі числа BigInteger і BigDecimal. Робота з датами. LocalDate. Виключення (exceptions). Виключна ситуація. Ієрархія виключень. Ключові слова try, catch, finally. Оператори throw, throws. Основні типи

	виключень в Java. Класи Java для обробки виключних ситуацій з пакету java.lang
5	Тема 2.4. ПОТОКИ ВВЕДЕННЯ-ВИВЕДЕННЯ. РОБОТА З ФАЙЛАМИ. Потoki введення-виведення. Читання і запис файлів. FileInputStream і FileOutputStream. Закриття потоків. Класи ByteArrayInputStream і ByteArrayOutputStream. Буферизовані потоки BufferedInputStream і BufferedOutputStream. Форматування введення і виведення. PrintStream і PrintWriter. Класи DataOutputStream і DataInputStream. Читання і запис текстових файлів. Буферизація символних потоків. BufferedReader і BufferedWriter. Сериалізація об'єктів. Клас File. Робота з файлами і каталогами. Робота з ZIP-архівами. Клас Console Тема 2.5. БАГАТОПОТОЧНЕ ПРОГРАМУВАННЯ. Клас Thread. Створення та виконання потоків. Завершення і переривання потоку. Синхронізація потоків. оператор synchronized. Методи wait і notify. Семафори. Обмін між потоками. Клас Exchanger. Клас Phaser. Блокування. ReentrantLock. Умови в блокуванні
6	Тема 3.1. СТВОРЕННЯ ГРАФІЧНОГО ІНТЕРФЕЙСУ КОРИСТУВАЧА НА JAVA. Бібліотека Swing. Бібліотеки для створення графічного інтерфейсу. Анонімні вкладені класи. Інтерфейс Runnable. Використання бібліотеки Swing. Вікна верхнього рівня. Клас JFrame. Малювання у вікні. Особливості малювання в класах Component і JComponent. Обчислення області малювання. Обробка подій.
7	Тема 3.2. АПЛЕТИ В JAVA. Поняття аплету (Applets). Кросплатформенність. Життєвий цикл. Виконання аплету. Клас JApplet.
8	Тема 3.3. РОЗРОБКА WEB-АПЛІКАЦІЙ НА JAVA. Принципи роботи веб-аплікацій. Фреймворки для розробки веб-аплікацій. Приклади розробки MVC архітектури. Технологія Servlets API. Клас HttpServlet. API-інтерфейси пакету javax.servlet.http. Обробка Http – запитів. Контейнер сервлетів Tomcat. Способи побудови веб-сторінок. Розгортання веб-аплікацій. War-архіви та запуск веб-аплікацій. Конфігурування за допомогою web.xml та анотацій. Використання Maven для побудови аплікації Тема 3.4. ANDROID-ПРОГРАМУВАННЯ. Початок роботи з Android. Основи створення інтерфейсу. Основні елементи управління. Ресурси. Activity. Робота з зображеннями. Адаптери і списки. Ситі і теми. Меню. Фрагменти. Багатопоточність і асинхронність. Робота з мережею. WebView. Робота з мультимедіа. Налаштування і стан додатку. Робота з файловою системою. Робота з базами даних SQLite. Перегортання сторінок і ViewPager. Сервіси. Телефонія і комунікація. Діалогові вікна. Анімація. Провайдери контенту. Публікація додатків. Робота з XML. Робота з Github
9	Тема 4.1. АНАЛІЗ ЗАСОБІВ РЕАЛІЗАЦІЇ ООП В МОВІ C++. Мотивація створення ООП, альтернативні підходи, тонкощі використання. Реалізація інкапсуляції та принципу DRY з та без ООП в C++
10	Тема 4.2. КЛАС ТА ОБ'ЄКТ В C++. Поля та методи класу. Варіанти синтаксису та структури опису класу. Інкапсуляція через специфікатори доступу. Звернення до об'єкту через вказівник. Конструктор класу. Стандартні списки ініціалізації. Деструктори. Конструктор копіювання. Конструктор переносу.
11	Тема 4.3. УСПАДКУВАННЯ ЯК КЛЮЧОВИЙ МОМЕНТ ООП. Сценарії використання. Недоліки та критика. Альтернативи. Порядок виклику конструкторів та деструкторів при успадкуванні. Делегування та успадкування конструкторів. Множинне успадкування.
12	Тема 4.4. РЕАЛІЗАЦІЯ ПОЛІМОРФІЗМУ В C++. Пропуски в ієрархії. Абстрактні класи. Концепція інтерфейсу в ООП. Віртуальні деструктори. Обробка виключень. Виключення-об'єкти, варіанти перехоплення, використання поліморфізму. Обмеження

13	Тема 5.1. ЗНАЙОМСТВО З C#.NET. Історія, базова архітектура, переваги та недоліки. Базовий синтаксис C# - вбудовані типи, змінні, керуючі конструкції, масиви, методи, структури, типи-посилання та типи-значення.
14	Тема 5.2. ОСНОВИ РЕАЛІЗАЦІЇ ООП в C#. Базовий синтаксис класів. Статичні члени класу. Інкапсуляція, успадкування, поліморфізм в C#. Інтерфейси як заміна множинному успадкуванню. Обробка виключень.
15	Тема 5.3. ТЕХНОЛОГІЇ ІНТЕГРАЦІЇ ЗАПИТІВ В МОВІ C# (LINQ). Ідея, переваги та недоліки. Сценарії використання. Автоматичне збирання сміття. Ідея, переваги та недоліки. Можливість впливу на garbage collector
16	Тема 5.4. РОЗРОБКА ВЕБ-ДОДАТКІВ З ЗАСТОСУВАННЯМ C#.Net. Розробка веб-додатків як одна з сфер застосування C#.Net. Підхід MVC, платформа ASP.NET MVC

5. ТЕМИ ЛАБОРАТОРНО-ПРАКТИЧНИХ ЗАНЯТЬ

№ з/п	Назва теми	Кількість годин
1.	Розробка та налагодження програм роботи з рядками	4
2.	Розробка та налагодження програм обробки лямбда-виразів	4
3.	Розробка та налагодження програм обробки об'єктних типів даних	2
4.	Розробка та налагодження програм обробки виключень та регулярних виразів	2
5.	Розробка та налагодження програм роботи з файлами	4
6.	Розробка та налагодження багатопоточних програм	2
7	Розробка та налагодження програм створення графічного інтерфейсу на Java	2
8	Розробка та налагодження аплетів в Java	4
9	Розробка та налагодження програм web-аплікацій на Java	4
10	Розробка та налагодження програм в середовищі Android	4
11	Розробка та налагодження програм реалізації інкапсуляції та принципу DRY в C++	4
12	Розробка та налагодження програм на C++ з використанням синтаксису та структури опису класу	4
13	Розробка та налагодження програм на C++ з використанням синтаксису успадкування	4
14	Розробка та налагодження програм на C++ з використанням синтаксису поліморфізму	4
15	Розробка та налагодження програм на основі базового синтаксису C#	4
16	Розробка та налагодження програм обробки об'єктних типів даних на основі синтаксису C#	4
17	Розробка та налагодження програм на основі технології LINQ	4
18	Розробка та налагодження веб-додатків з застосуванням C#.Net	4
Разом:		64

6. САМОСТІЙНА РОБОТА

№ з/п	Назва теми	Кількість годин
-------	------------	-----------------

1.	Рекурсивні функції. Обробка виняткових ситуацій.	6
2.	Приведення до потрібного реєстру символів. Методи toLowerCase() та toUpperCase(). Перетворення масиву символів char[] в рядок String. Метод copyValueOf(). Перетворення числа в рядок. Метод valueOf().	6
3.	Спадкування і узагальнення. Довідкові типи і копіювання об'єктів.	6
4.	Можливості застосування Regular Expressions. Звичайні символи (літера-ли) і спеціальні символи (метасимволи). Символьні класи (набори символів). Групи. Переліки. Квантифікація (пошук послідовностей). Групування	6
5.	Робота с ZIP-архивами. Клас Console	6
6.	Обчислення області малювання. Клас Graphics. Клас Color. Клас Font. Скроллінг. Клас JScrollPane.	6
7.	Створення Web-сторінки. Тестування у програмі AppletViewer	6
8.	Діалогові вікна. Анімація. Провайдери контенту. Публікація додатків. Робота з XML. Робота з Github	6
9.	Перезавантаження операторів. Варіанти, нюанси використання. Особливості для операторів =, (), [], <<, >>.	9
10.	Дружні функції, дружні класи. Вкладені класи. Переваги і недоліки.	9
11.	Повтор поняття поліморфізму, переваги та недоліки, альтернативи. Віртуальні функції як основа поліморфізму в C++.	9
12.	Поняття шаблону в C++ (template). Мета, переваги та недоліки, приклади	9
13.	Масиви, методи, структури, типи-посилання та типи-значення	9
14.	Узагальнення (generics). Колекції. Делегати та лямбда-вирази	9
15.	Автоматичне збирання сміття. Ідея, переваги та недоліки	9
16.	Підхід MVC, платформа ASP.NET MVC	9
Разом:		144

7. МЕТОДИ НАВЧАННЯ

При викладанні навчальної дисципліни «Об'єктно-орієнтоване програмування» застосовуються інформаційні та практичні методи навчання: класичні лекції, лекції-дискусії та лабораторні заняття, а також консультації з виконання самостійної роботи студентів, письмові завдання при проведенні контрольних робіт.

Методи навчально-пізнавальної діяльності: пояснювально-ілюстративний метод, репродуктивний метод, метод проблемного викладу, частково-пошуковий або евристичний метод, дослідницький метод.

Методи стимулювання й мотивації навчально-пізнавальної діяльності: індуктивні і дедуктивні методи навчання, методи стимулювання і мотивації навчання.

8. МЕТОДИ КОНТРОЛЮ

Відповідно до плану вивчення дисципліни «Об'єктно-орієнтоване програмування» передбачається проведення поточного та підсумкового контролю.

Поточний контроль – оцінювання рівня знань, умінь та навичок осіб, які навчаються, що здійснюється в ході навчального процесу шляхом проведення письмового опитування по закінченню розділів (модульний колоквиум). Модульний контроль при особливих ситуаціях може проводитись у формі мережевого комп'ютерного тесту з фіксованим часом відповіді.

9. ФОРМА ПІДСУМКОВОГО КОНТРОЛЮ УСПІШНОСТІ НАВЧАННЯ

Формами підсумкового контролю є залік (у першому семестрі) та **екзамен**, які складаються очно (при особливій ситуації – у формі комп'ютерного тесту) в період призначений деканатом або за індивідуальним графіком, який затверджується навчальним планом.

10. СХЕМА НАРАХУВАННЯ ТА РОЗПОДІЛ БАЛІВ, ЯКІ ОТРИМУЮТЬ СТУДЕНТИ

Накопичення балів протягом семестру

№ з/п	Вид діяльності	Кількість балів за дидактичну одиницю	Кількість	Загальна кількість балів
1	Тестування за матеріалами лекцій	3	8	24
2	Виконання лабораторних робіт	4	8	32
3	Виконання самостійних робіт	1	4	4
Залік (Екзамен)		40		40
Максимальна оцінка				100

Загальна оцінка знань студентів за поточним контролем

Результати поточного контролю знань студентів в цілому оцінюються в діапазоні від **0** до **60** балів.

Студент допускається до підсумкового контролю за умови виконання вимог навчальної програми та у разі, якщо за поточну навчальну діяльність він набрав не менше **36** балів.

Підсумкове оцінювання знань студентів

Підсумкове оцінювання знань студентів проводиться у формі **екзамену**.

Критерії оцінювання знань під час іспиту

Максимальна кількість балів, яку можна отримати на екзамені складає **40** балів.

Розподіл балів оцінювання при підсумковому контролі з навчальної дисципліни

Оцінка в балах за поточне оцінювання	Оцінка в балах за підсумкове оцінювання	Оцінка за національною шкалою
54-60	36-40	Відмінно
45-53	30-35	Добре
36-44	24-29	Задовільно
менше 36	менше 24	Незадовільно

Під час оцінювання відповіді на окреме питання додатково враховуються допущені недоліки та помилки, якими вважаються:

- неохайне оформлення роботи (не загальноприйняті скорочення, незрозумілий почерк, використання олівців замість чітких чорнил) (мінус 2 бали);

- неточності в назвах окремих термінів та понять (мінус 4 бали).

Критерії оцінювання відповіді на теоретичні питання білету:

1. Повна відповідь на питання, яка оцінюється **«відмінно»**, повинна відповідати таким вимогам:

- розгорнутий, вичерпний виклад змісту даної у питанні проблеми;
- повний перелік необхідних для розкриття змісту питання термінів та положень;

- здатність здійснювати порівняльний аналіз різних систем та самостійно робити логічні висновки й узагальнення;

- уміння користуватись методами наукового аналізу;

- демонстрація здатності висловлення та аргументування власного ставлення до альтернативних поглядів на дане питання;

2. Відповідь на питання оцінюється **«добре»**, якщо:

- відносно відповіді на найвищий бал не зроблено розкриття хоча б одного з пунктів, вказаних вище (якщо він явно потрібний для вичерпного розкриття питання) або, якщо:

- при розкритті змісту питання в цілому правильно за зазначеними вимогами зроблені окремі помилки під час: використання формул.

3. Відповідь на питання оцінюється **«задовільно»**, якщо:

- відносно відповіді на найвищий бал не зроблено розкриття чотирьох чи більше пунктів, зазначених у вимогах до нього (якщо вони явно потрібні для вичерпного розкриття питання);

- одночасно присутні чотири чи більше типів недоліків, які окремо характеризують критерій оцінки питання;

- висновки, зроблені під час відповіді, не відповідають правильним чи загально визначеним при відсутності доказів супротивного аргументами, зазначеними у відповіді;

- характер відповіді дає підставу стверджувати, що особа, яка складає іспит, не зовсім правильно зрозуміла зміст питання чи не знає правильної відповіді і тому не відповіла на нього по суті, допустивши грубі помилки у змісті відповіді.

З урахуванням вищевикладеного результати іспиту оцінюються в діапазоні від **0** до **40** балів для студентів.

Загальна підсумкова оцінка з дисципліни складається з суми балів за результати поточного контролю знань та за виконання завдань, що виносяться на іспит.

Загальна підсумкова оцінка не може перевищувати **100 балів**.

Загальна підсумкова оцінка в балах, за національною шкалою та за шкалою ECTS заноситься до заліково-екзаменаційної відомості, навчальної картки та залікової книжки студента.

Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проекту	для заліку
90-100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
66-73	D	задовільно	
60-65	E		
30-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
1-29	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

11. МЕТОДИЧНЕ ЗАБЕЗПЕЧЕННЯ

- робоча навчальна програма дисципліни;
- електронний курс у MOODLE з тезами лекцій, інструкціями до лабораторних занять, тестами та матеріалами для самостійної роботи студентів;
- перелік питань до екзамену.

12. РЕКОМЕНДОВАНА ЛІТЕРАТУРА

1. Эккель Б. Философия Java. 4-е изд. Пер. з англ. Питер, 2011. 640 с.
2. Хорстманн, К. С., Корнелл, Г. Библиотека профессионала. Java 2 : Том 1. Основы. – 8-е изд. Пер. з англ. Вильямс, 2013. 816 с.
3. Перри, Б. У. Java сервлеты и JSP. Сборник рецептов. Пер. з англ. Кудиц-пресс, 2009. 768 с.
4. Шилдт Г. Полный справочник по C++, 4-е изд. Пер. з англ. Издательский дом “Вильямс”, 2006.
5. Кнут Д. Искусство программирования. Т. 1-3, 3-е изд. Пер. з англ. Издательский дом “Вильямс”, 2000.
6. Троелсен Э. Язык программирования C# 5.0 и платформа .NET 4.5, 6-е изд. Пер. з англ. Издательский дом “Вильямс”, 2013.
6. Шилдт Г. Полное руководство C# 4.0. Пер. з англ. Издательский дом “Вильямс”, 2011.

7. Буч Г. Объектно-ориентированный анализ и проектирование с примерами приложений на C++, 2-е изд., пер. с англ. "Издательство Бином", СПб.: "Невский диалект", 1999

Интернет ресурсы

1. <https://mva.microsoft.com/ru/training-courses/-c-1-8669?l=MAuqZiG14404984382>
5. <https://mva.microsoft.com/ru/training-courses/-c-8622?l=DQMesDF17804984382>
6. <https://codeasy.net>
<https://docs.microsoft.com/en-us/dotnet/csharp/index>