

Маріупольський державний університет

Кафедра _____ Математичних методів та системного аналізу _____



Т.В. Шабельник

_____ р.

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

_____ ВДПП 2.2.8 «Крос-платформне програмування» _____

(шифр і назва навчальної дисципліни)

спеціальність 125 – Кібербезпека _____

(шифр і назва спеціальності)

спеціалізація _____ кібербезпека _____

(назва спеціалізації)

факультет _____ економіко-правовий _____

(назва факультету)

2020-2021 рік

Робоча програма Крос-платформне програмування для студентів за напрямом
(назва навчальної дисципліни)
підготовки 125 - Кібербезпека

Розробник:

Гранкін Д.В. кандидат фізико-математичних наук, доцент
(вказати авторів, їхні посади, наукові ступені та вчені звання)

Робоча програма затверджена на засіданні кафедри математичних методів та системного аналізу

Протокол від “27” серпня 2020 року № 1

Завідувач кафедри математичних методів та системного аналізу



(Шабельник Т.В.)
(прізвище та ініціали)

“27” серпня 2020 року

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика навчальної дисципліни	
		денна форма навчання	заочна форма навчання
Кількість кредитів – 6	Галузь знань <u>12 Інформаційні технології</u> (шифр і назва)	Нормативна	
	Напрямок підготовки <u>125 Кібербезпека</u> (шифр і назва)		
Модулів – 1	Спеціальність (професійне спрямування): _____	Рік підготовки:	
Змістових модулів – 3		4-й	
Індивідуальне науково-дослідне завдання: <u>вирішення типових завдань за темами змістовних модулів</u> (назва)		Семестр	
Загальна кількість годин – 180		7-й	
		Лекції	
Тижневих годин для денної форми навчання: аудиторних – 2 самостійної роботи студента – 8	Освітньо-кваліфікаційний рівень: бакалавр	16	6
		Практичні, семінарські	
		16	6
		Лабораторні	
		Самостійна робота	
		148	168
		Індивідуальні завдання: год.	
		Вид контролю:	
екзамен			

Примітка:

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:

для денної форми навчання – 22%,

для заочної форми навчання – 7%.

2. Мета та завдання навчальної дисципліни

Мета вивчення курсу: формування та узагальнення спеціальних знань та навичок студентів з питань створення крос-платформних прикладень, зокрема на мові програмування Java. Навчити студентів правильно проводити аналіз інформаційної безпеки веб-орієнтованих додатків та самостійно реалізовувати їх з використанням платформи Java та споріднених фреймворків.

Завданнями курсу є:

- ознайомлення студентів з особливостями мови програмування Java;

- ознайомлення студентів з сучасними методологіями розробки програмного забезпечення із акцентуванням тих, що використовуються в екосистемі Java;
- ознайомлення студентів з основними бібліотеками та сучасними інструментами, що використовуються у проектах, які розроблюються на Java;
- вивчення синтаксису та основоположних принципів ядра мови (Java Core);
- вивчення принципів і підходів, що використовуються при створенні прикладень на Java, у яких реалізується багатопоточність;
- ознайомлення з технологіями для створення веб-прикладень;
- освоєння студентами основних технологічних методів практичного застосування мовних засобів програмування для розробки крос-платформного програмного продукту.

У результаті вивчення навчальної дисципліни студент повинен

знати:

- предмет та головні поняття курсу;
- стандартні інструменти JDK;
- Core Java;
- фреймворк Collections;
- підходи до створення багатонитевих застосунків на Java;
- інструменти для збірки Java-проекту;
- підходи, що використовуються для тестування прикладень на Java;
- основні патерни для створення веб-застосунків.

вміти:

- створювати робочі крос-платформні застосунки на Java;
- використовувати принципи об'єктно-орієнтованого програмування (ООП) та ООП-патерни для створення крос-платформних застосунків на Java;
- використовувати сучасний інструментарій розробника і бібліотеки, зокрема системи контролю версій (Git), збірки проекту (Apache Maven), бібліотеки для модульного тестування (JUnit), контейнер сервлетів / веб-сервер (Apache Tomcat).

Місце навчальної дисципліни в освітній програмі: ВК 12

Передумови для вивчення дисципліни: Алгоритми та структури даних, Програмування, Архітектура комп'ютерних систем, Комп'ютерні мережі, Інформаційні технології та системи.

Результати навчання:

РН2 - організувати власну професійну діяльність, обирати оптимальні методи та способи розв'язування складних спеціалізованих задач та практичних проблем у професійній діяльності, оцінювати їхню ефективність;

РН3 - використовувати результати самостійного пошуку, аналізу та синтезу інформації з різних джерел для ефективного рішення спеціалізованих задач професійної діяльності;

РН11 - виконувати аналіз зв'язків між інформаційними процесами на віддалених обчислювальних системах;

РН14 - вирішувати завдання захисту програм та інформації, що обробляється в інформаційно-телекомунікаційних системах програмно-апаратними засобами та давати оцінку результативності якості прийнятих рішень;

РН 15 - використовувати сучасне програмно-апаратне забезпечення інформаційно-телекомунікаційних технологій;

РН 19 - застосовувати теорії та методи захисту для забезпечення безпеки інформації в інформаційно-телекомунікаційних системах;

РН21 - вирішувати задачі забезпечення та супроводу (в.т. числі: огляд, тестування, підзвітність) системи управління доступом згідно встановленої політики безпеки в інформаційно-телекомунікаційних (автоматизованих) системах;

RH22 - вирішувати задачі управління процедурами ідентифікації, автентифікації, авторизації процесів і користувачів в інформаційно-телекомунікаційних системах згідно встановленої політики інформаційної та /або кібербезпеки;

RH25 - забезпечувати введення підзвітності системи управління доступом до електронних інформаційних ресурсів і процесів в інформаційних та інформаційно-телекомунікаційних (автоматизованих) системах з використанням журналів реєстрації подій, їх аналізу та встановлених процедур захисту;

RH53 - вирішувати задачі аналізу програмного коду на наявність можливих загроз.

3. Програма навчальної дисципліни

Змістовий модуль 1. Ядро мови Java (JavaCore).

Тема 1. Вступ. Огляд мови Java. Методології розробки програмного забезпечення. Огляд інструментів і технологій.

Основні особливості мови. Застосування платформи Java. Історія створення. Класифікація платформ Java: Java SE/EE/ME/Card. Комплект розробника JDK. Середовище виконання JRE. Віртуальна машина JVM. Інші мови, що використовують JVM. JIT-компіляція. ПЮВЕ-індекс мови Java. Причини популярності мови. Порівняння версій Java 1.6, 1.7 і 1.8. Ключові нововведення в Java 9-15. Каскадна модель (Waterfall). Розробка через тестування (V-model). Інкрементна модель (Incremental). Модель швидкої розробки додатків (RAD). Ітеративна модель (Iterative). Спіральна модель (Spiral). Гнучка методологія розробки (Agile). Agile.Kanban. Agile.Scrum: ключові ролі; спринт; беклог. Життєвий цикл спринту. Планування спринту. Зупинка спринту (Sprint Abnormal Termination). Щоденний Scrum Meeting. Діаграма згоряння завдань (Burndown chart). Ретроспектива. Різниця між Kanban і Scrum. Оболонка Bash. Системи контролю версій. Репозиторії. Огляд найбільш популярних інструментів і технологій Java/Java Enterprise. JUnit — бібліотека для модульного тестування програмного забезпечення. Git — розподілена система керування версіями. Jenkins — програмна система для забезпечення процесу безперервної інтеграції програмного забезпечення. Hibernate — реалізація специфікації JPA (Java Persistence API) для вирішення завдань об'єктно-реляційного відображення. Apache Maven — фреймворк для автоматизації збірки проєктів. Контейнер сервлетів Apache Tomcat. СУБД MySQL. Об'єктно-реляційна СУБД PostgreSQL. Фреймворк Spring і його модулі. Інтегроване середовище розробника IntelliJ IDEA.

Тема 2. Системи управління версіями. Принципи об'єктно-орієнтованого програмування (ООП). Структура пам'яті.

Поняття і призначення систем управління версіями (VCS). Порівняння різних VCS (Git, SVN, Mercurial). Історія Git. Установка і початкова настройка Git. Створення Git-репозиторія. Запис змін в репозиторій. Перегляд історії комітів. Операції скасування. Робота з віддаленими репозиторіями. Інтеграція з GitHub, BitBucket, GitLab. Робота з мітками. Псевдоніми в Git. Розгалуження і зліття в Git. Управління та робота з гілками. Віддалені гілки. Перебазування. Git на сервері. Розподілений Git. Інструменти Git. Методологія процедурно-орієнтованого і об'єктно-орієнтованого програмування. Об'єкти (стан, поведінка, унікальність). Класи. Інкапсуляція. Спадкування. Поліморфізм. Типи відносин між класами. Спадкування (узагальнення), агрегація, композиція, асоціація. Переваги та недоліки ООП. Куча, стек, регістри, константи. Прибирання сміття в Java (Java garbage collection).

Тема 3. Типи даних. Пакети. Об'єктна модель в Java. Базові елементи Java Core.

Примітивні і посилальні типи даних. Об'єкти і правила роботи з ними. Імена. Пакети. Модифікатори доступу. Область видимості. Угоди по іменуванню. Пакет java.lang. Клас Object. Методи класу Object. Клас String. Клас Class. Класи-обгортки. Перераховувані типи (enum) в Java. Модифікатор static. Клас Object: контракт equals/hashCode. Інтерфейси, абстрактні класи, поліморфізм. Відмінності абстрактного класу від інтерфейсу. Патерн проектування "Шаблонний метод". Конструктори класів. Ключові слова this, super. Патерн проектування "Сінглтон". Метод main. Типовий "Hello World"-додаток. Робота з Java в командному рядку. Етапи компіляції і запуску. Ручна компіляція і запуск програми. Байт-код. Відображення байт-коду класу, javap. Реалізація типових ал-

горитмічних структур на Java. Розгалуження: if, if else. Цикли: for, while, do while. Ініціалізація полів в Java.

Тема 4. Робота зі строками. Винятки (Exceptions). Рефлексія в Java. Анотації. Модульне тестування.

Строки в Java. Кодування символів. Управління строками, функції для роботи зі строками в Java. StringBuilder і StringBuffer. String literal pool. Причини виникнення винятку. Клас Throwable. Ієрархія винятків. Винятки, що перевіряються та ті, що не перевіряються (checked/unchecked). Методи винятків. Обробка винятків — try і catch. Перехоплення багатотипних винятків. Ключові слова throws/throw. Блок finally. Конструкція try-with-resources. Створення своїх власних винятків. Анотації в Java. Java Reflection API. Обмеження при роботі з рефлексією в Java. Java Reflection: Робота з класами; реалізовані інтерфейси; конструктори; поля; методи; приватні поля і методи; анотації; дженерики; масиви. Тестування коду Java за допомогою фреймворку для модульного тестування JUnit.

Змістовий модуль 2. Розширені засоби крос-платформного програмування на Java.

Тема 5. Контейнери/колекції. Java Collections Framework.

Архітектура. Інтерфейси List, Set, Map, Queue, Iterator, ListIterator. Структури даних ArrayList, HashSet, TreeSet, LinkedHashSet, LinkedList; HashMap, Hashtable, TreeMap, LinkedHashMap. Методи інтерфейсу Collection. Клас Collections. Iterator/Iterable. Вкладені і внутрішні класи — приклади в Collections API: Arrays.asList, ArrayList.iterator. Локальні і анонімні класи. Comparator.

Тема 6. Узагальнення/Дженерики (Generic).

Узагальнене програмування. Алмазний синтаксис (Diamond syntax). Універсальні методи (Generic methods). Wildcards (Маски). Обмеження дженериків.

Тема 7. Журналювання (logging). Робота з датами і часом.

Журналювання в Java. Типи логів і вимоги до них. Уніфіковані системи логування. Apache Log4J. Java Logging API (пакет java.util.logging). Уніфікований інтерфейс Apache Commons Logging. Simple Logging Facade for Java (SLF4J). Використання Log4J: принципи і поняття; конфігурація; використання в програмному коді; управління під час виконання. Класи Date, Calendar, TimeZone. Перетворення дати і часу між часовими зонами в Java. Joda Time library. Java 8 Date API.

Тема 8. Робота з файлами і ресурсами. Введення/виведення. Серіалізація. NIO (Non-blocking I/O, New I/O). Java 8 Stream API.

Клас File. Робота з файловою системою. Робота з ресурсами. Найкращі практики щодо звільнення ресурсів в Java: підхід try-catch-finally і його недоліки; використання com.google.common.io.Closer з бібліотеки Google Guava; використання try-with-resources. Пакет java.io. Патерн проектування "Декоратор". Класи Reader і Writer і їх спадкоємці. Серіалізація об'єктів (serialization). NIO: API для реалізації високопродуктивних операцій введення-виведення. Ключові особливості NIO: канали та селектори, буфери, кодування символів. Використання інтерфейсу Path. Використання класу Files. Копіювання, переміщення і видалення файлів. Обхід дерева файлів. Пошук файлів. Відстеження змін в каталозі. Основи Java 8 Stream API (java.util.Stream). Фільтрація даних. Сортуння даних. Сімейство методів Match. Метод count(). Еволюція Java I/O API на прикладі читання рядків з файлу.

Змістовий модуль 3. Сучасні підходи та інструменти у розробці крос-платформних прикладень на Java.

Тема 9. Формат XML. Робота з XML в Java. JSON (JavaScript Object Notation). Багатонитевість. Паралельне виконання.

XML формат і технології. Робота з XML в Java. XSL (eXtensible Stylesheet Language) — сімейство рекомендацій консорціуму W3C, яке описує мови перетворення і візуалізації XML-документів. Об'єктна модель документа (DOM). Спосіб послідовного читання/запису XML-файлів SAX

(Simple API for XML). Streaming API for XML (StAX). Конвертація XML-схем в класові уявлення за допомогою JAXB (Java Architecture for XML Binding). Реалізація зберігання в XML. Текстовий формат обміну даними JSON. Серіалізація складних структур: порівняння XML і JSON підходів. Використання JSON в веб-додатках для обміну даними між браузером і сервером (AJAX) і між серверами (програмні HTTP-сполучення). JSON в JavaScript. Архітектурний стиль взаємодії компонентів розподіленого мережного застосунку REST (Representational State Transfer). Робота з JSON в Java: Google GSON і Jackson. Реалізація зберігання в JSON. DataInputStream/DataOutputStream. Закон Мура. Закон Амдала. Потоки виконання. Базові класи для роботи з потоками. Клас Thread. Інтерфейс Runnable. Робота з пріоритетами. Daemons. Синхронізація. Блокування. Методи wait(), notify(), notifyAll() класу Object. Лінива ініціалізація, JMM (Java Memory Model). Потокобезпечна реалізація Singleton в Java. Блокування з подвійною перевіркою (Double checked locking). Модифікатори final, volatile. Патерн "Ініціалізація на вимогу власника" (initialization-on-demand holder idiom). Взаємне блокування (deadlock). Пакет java.util.concurrent.*. Compare-and-swap. Синхронізатори. Використання ThreadLocal-змінних.

Тема 10. Робота з базою даних в Java-проекті. Технології Java для розробки Web-застосунків. DB-Engines Ranking. Реляційна СУБД. Порівняння реляційних і NoSQL баз даних. SQL. Денормалізація. Primary (PK) і Foreign Key (FK), Cascade. PostgreSQL. Надійність. Робота з базами даних з IntelliJ IDEA. IDEA Database tools. Робота з файлами властивостей Java Properties. Конфігурація БД і каталогу зберігання. Підключення БД в проект. JDBC (Java DataBase Connectivity — з'єднання з базами даних на Java); пакет java.sql. Архітектура JDBC. Робота з базами даних за допомогою JDBC драйвера. SQL об'єднання JOIN: LEFT/RIGHT/INNER/OUTER. Транзакція. Вимоги ACID (Атомарність/Atomicity, Узгодженість/Consistency, Ізольованість/Isolation, Стійкість/Durability). Рівні ізоляції транзакцій. Протокол HTTP. Інструменти розробника в Google Chrome. Дескриптор розгортання. web.xml. Побудова/структура WAR. Статичні ресурси. Налаштування та деплой в Tomcat. Tomcat manager. Запуск Tomcat з IntelliJ IDEA. Динамічне оновлення без передеплою. Сервлети. Створення сервлету. Параметри. Кодування. JSP (JavaServer Pages) і бібліотеки custom-тегів. Заздалегідь визначені змінні в JSP. Стандартні елементи action. Використання відносних шляхів в JSP. Відмінність Redirect від Forward. JSTL — стандартна бібліотека тегів JSP (JavaServer Pages Standard Tag Library). JSTL для написання JSP сторінок. Розгортання Web-додатку в Heroku CLI. Завантаження класів в Java. Apache Tomcat Class Loader.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин											
	денна форма						Заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб	інд	с.р.		л	п	лаб	інд	с.р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовий модуль 1. Ядро мови Java (JavaCore).												
Тема 1. Вступ. Огляд мови Java. Методології розробки програмного забезпечення. Огляд інструментів і технологій.	15	1				14	16.5	0.5				16
Тема 2. Системи управління версіями. Принципи об'єктно-орієнтованого програмування (ООП). Структура пам'яті.	16	1				15	16.5	0.5				16
Тема 3. Типи даних. Пакети. Об'єктна модель в Java. Базові елементи Java Core.	19	2	2			15	18	1				17
Тема 4. Робота зі строками. Винятки (Exceptions). Рефлексія в Java. Анотації. Модульне тестування.	18	1	2			15	17.5	0.5				17
Разом за модулем 1	68	5	4			59	68.5	2.5				66
Змістовий модуль 2. Розширені засоби крос-платформного програмування на Java.												
Тема 5. Контейнери/колекції. Java Collections Framework.	19	2	2			15	18.5	0.5	1			17
Тема 6. Узагальнення/Дженерики (Generic).	19	2	2			15	18.5	0.5	1			17
Тема 7. Журналювання (logging). Робота з датами і часом.	15	1	2			12	16.5	0.5	1			15
Тема 8. Робота з файлами і ресурсами. Введення/виведення. Сериалізація. NIO (Non-blocking I/O, New I/O). Java 8 Stream API.	21	2	2			17	18.5	0.5	1			17
Разом за модулем 2	74	7	8			59	72	2	4			66
Змістовий модуль 3. Сучасні підходи та інструменти у розробці крос-платформних прикладень на Java.												
Тема 9. Формат XML. Робота з XML в Java. JSON (JavaScript Object Notation). Багатонитевість. Паралельне виконання.	19	2	2			15	19.5	0.5	1			18
Тема 10. Робота з базою даних в Java-проекті. Технології Java для розробки Web-застосунків.	19	2	2			15	20	1	1			18
Разом за модулем 3	38	4	4			30	39.5	1.5	2			36
Усього годин	180	16	16			148	180	6	6			168

5. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1	Підготовка робочого оточення.	2
2	Реалізація найпростіших застосунків на Java.	2
3	Створення застосунків з об'єктно-орієнтованою архітектурою.	2
4	Створення віконних застосунків з використанням бібліотеки Swing.	2
5	Робота з масивами.	2
6	Робота з колекціями.	2
7	Робота з потоками введення/виведення. Серіалізація.	2
8	Робота з нитками (Multithreading).	2
Усього		16

6. Самостійна робота

№ з/п	Назва теми	Кількість годин
1	Сортування в Java. Інтерфейси Comparable і Comparator.	14
2	Java 8 Stream API (java.util.Stream).	14
3	Робота з JSON в Java: Google GSON і Jackson. Реалізація зберігання в JSON.	15
4	Багатонитевість в Java (Multithreading). Пакет java.util.concurrent.*.	15
5	JSTL. Стандартна бібліотека тегів JSP. Використання для написання JSP сторінок.	15
6	Персистентність. Фреймворк Hibernate.	15
7	Java EE. Специфікація технології написання і підтримки серверних компонентів Enterprise JavaBeans.	15
8	Архітектура (патерн) MVC.	15
9	Мікросервісна архітектура.	15
10	Використання Spring Framework у Java-проектах.	15
Усього		148

7. Методи навчання: лекції, практичні заняття, самостійне вивчення деяких теоретичних питань, самостійна робота студентів з виконанням практичних завдань.

8. Критерії оцінювання

Критерії поточного оцінювання знань студентів.

Усний виступ та виконання письмового завдання, тестування	Критерії оцінювання
5	В повному обсязі володіє навчальним матеріалом, вільно самостійно та аргументовано його викладає під час усних виступів та письмових відповідей, глибоко та всебічно розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому обов'язкову та додаткову літературу. Правильно вирішив усі тестові завдання.
4	Достатньо повно володіє навчальним матеріалом, обґрунтовано його викладає під час усних виступів та письмових відповідей, в основному розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому обов'язкову літературу. Але при викладанні деяких питань не вистачає достатньої глибини та аргументації, допускає

	ються при цьому окремі несуттєві неточності та незначні помилки. Правильно вирішив більшість тестових завдань.
3	В цілому володіє навчальним матеріалом викладає його основний зміст під час усних виступів та письмових відповідей, але без глибокого всебічного аналізу, обґрунтування та аргументації, без використання необхідної літератури допускаючи при цьому окремі суттєві неточності та помилки. Правильно вирішив половину тестових завдань.
2	Не в повному обсязі володіє навчальним матеріалом. Фрагментарно, поверхово (без аргументації обґрунтування) викладає його під час усних виступів та письмових відповідей, недостатньо розкриває зміст теоретичних питань та практичних завдань, допускаючи при цьому суттєві неточності, правильно вирішив меншість тестових завдань.
1	Частково володіє навчальним матеріалом не в змозі викласти зміст більшості питань теми під час усних виступів та письмових відповідей, допускаючи при цьому суттєві помилки. Правильно вирішив окремі тестові завдання.

Доповнення виступу:

2 бали – отримують студенти, які глибоко володіють матеріалом, чітко визначили його зміст; зробили глибокий системний аналіз змісту виступу, виявили нові ідеї та положення, що не були розглянуті, але суттєво впливають на зміст доповіді, надали власні аргументи щодо основних положень даної теми.

1 бал отримують студенти, які виклали матеріал з обговорюваної теми, що доповнює зміст виступу, поглиблює знання з цієї теми та висловили власну думку.

Суттєві запитання до доповідачів:

2 бали отримують студенти, які своїм запитанням до виступаючого суттєво і конструктивно можуть доповнити хід обговорення теми.

1 бал отримують студенти, які у своєму запитанні до виступаючого вимагають додаткової інформації з ключових проблем теми, що розглядається.

Експрес-контроль:

2 бали нараховуються студентам, які вільно володіють усім навчальним матеріалом, орієнтуються в темі та аргументовано висловлюють свої думки.

1 бал отримують студенти, які частково володіють матеріалом та можуть окреслити лише деякі проблеми теми.

Складання словника основних термінів, що визначені програмою курсу (за темами):

Програмою курсу визначено перелік ключових термінів, що розкривають зміст кожної теми. Студентам пропонується скласти словник основних термінів з конкретної теми на останніх сторінках опорного конспекту лекцій.

2 бали нараховуються студентам, які не лише склали повний перелік визначених термінів з конкретної теми, а й можуть вільно розтлумачити їх зміст.

1 бал нараховуються студентам, які склали неповний перелік визначених термінів з конкретної теми і не можуть їх розтлумачити без конспекту.

Ведення опорного конспекту лекції:

2 бали нараховуються студентам, які в повному обсязі самостійно і творчо опрацювали всі питання лекції і вільно володіють її змістом.

1 бал нараховується студентам, які опрацювали лише окремі питання лекції і не достатньо вільно володіють її змістом.

Підготовка творчих завдань (есе, дайджест):

2 бали отримують студенти, які можуть виокремити з різних джерел основні положення, структурно об'єднати їх, коротко проаналізувати кожне з них та зробити ґрунтовні узагальнюючі висновки.

1 бал отримують студенти, які в цілому правильно виокремили основні положення кожного з джерел, але не зробили їх відповідного аналізу та узагальнюючих висновків.

Ведення конспекту першоджерел.

2 бали отримують студенти, які опрацювали всю необхідну обов'язкову літературу, засвоїли її основні теоретичні положення, вміють їх пояснити і розтлумачити.

1 бал отримують студенти, котрі опрацювали не всю необхідну літературу, не завжди розуміють її вихідні теоретичні положення, поверхово їх пояснюють.

Підсумковий модульний контроль знань студентів.
Критерії підсумкового модульного оцінювання знань студентів

Письмова контрольна робота або тестування	Критерії оцінювання
21-25	В повному обсязі володіє навчальним матеріалом, вільно самостійно та аргументовано його викладає під час усних виступів та письмових відповідей, глибоко та всебічно розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому обов'язкову та додаткову літературу. Правильно вирішив усі тестові завдання.
17-21	Достатньо повно володіє навчальним матеріалом, обґрунтовано його викладає під час усних виступів та письмових відповідей, в основному розкриває зміст теоретичних питань та практичних завдань, використовуючи при цьому обов'язкову літературу. Але при викладанні деяких питань не вистачає достатньої глибини та аргументації, допускаються при цьому окремі несуттєві неточності та незначні помилки. Правильно вирішив більшість тестових завдань.
14-17	В цілому володіє навчальним матеріалом викладає його основний зміст під час усних виступів та письмових відповідей, але без глибокого всебічного аналізу, обґрунтування та аргументації, без використання необхідної літератури допускаючи при цьому окремі суттєві неточності та помилки. Правильно вирішив половину тестових завдань.
10-14	Не в повному обсязі володіє навчальним матеріалом. Фрагментарно, поверхово (без аргументації та обґрунтування) викладає його під час усних виступів та письмових відповідей, недостатньо розкриває зміст теоретичних питань та практичних завдань, допускаючи при цьому суттєві неточності, правильно вирішив меншість тестових завдань.
10	Частково володіє навчальним матеріалом не в змозі викласти зміст більшості питань теми під час усних виступів та письмових відповідей, допускаючи при цьому суттєві помилки. Правильно вирішив окремі тестові завдання.
0	Не володіє навчальним матеріалом та не в змозі його викласти, не розуміє змісту теоретичних питань та практичних завдань. Не вирішив жодного тестового завдання.

9. Засоби оцінювання

Поточний контроль знань ЗВО здійснюється за допомогою тестів, опитувань по темах, захисту звітів про виконання практичних робіт. Модульний контроль здійснюється із застосуванням тестів або письмової контрольної роботи. Підсумковий контроль здійснюється у формі екзамену.

Завданням поточного контролю знань ЗВО є перевірка розуміння та запам'ятовування певного теоретичного матеріалу, умінь самостійної роботи зі спеціальною літературою, набуття практичних навичок роботи з ПК і окремими програмними засобами, вміння пояснити і захистити свою роботу.

10. Розподіл балів, які отримують студенти

Поточне тестування та самостійна робота													Сума (в балах)		
Змістовий модуль №1					Змістовий модуль №2					Змістовий модуль №3					
T1	T2	T3	T4	Тест	T5	T6	T7	T8	Тест	T9	T10	Тест	мод. контроль	екзамен	всього
2	2	2	2	2	5	5	5	5	5	5	5	5	50	50	100

T1, T2, ... – змістові теми

Шкала оцінювання: національна та ECTS

Сума балів за всі види навчальної діяльності	Оцінка ECTS	Оцінка за національною шкалою	
		для екзамену, курсового проєкту (роботи), практики	для заліку
90 – 100	A	відмінно	зараховано
82-89	B	добре	
74-81	C		
64-73	D	задовільно	
60-63	E		
35-59	FX	незадовільно з можливістю повторного складання	не зараховано з можливістю повторного складання
0-34	F	незадовільно з обов'язковим повторним вивченням дисципліни	не зараховано з обов'язковим повторним вивченням дисципліни

11. Рекомендовані джерела інформації

Основні:

1. Шилдт Г. Java 8. Полное руководство. – К.: Вильямс, 2015. – 1375 с.
2. Сьерра К., Бейтс Б. Изучаем Java. – Sebastopol, CA: O'Reilly, 2012. – 718 с.
3. Horstmann C.S. Core Java. Volume I—Fundamentals. – San Francisco: Prentice Hall, 2016. – 1007 p.
4. Horstmann C.S. Core Java. Volume II—Advanced Features. – San Francisco: Prentice Hall, 2017. – 2760 p.
5. Eckel B. Thinking in Java. – San Francisco: Prentice Hall, 2006. – 1057 p.
6. Horstmann C.S. Core Java. Big Java. Early Objects. – San Jose: Wiley, 2015. – 1444 p.
7. Bloch J. Effective Java. – San Francisco: Addison-Wesley, 2018. – 392 p.

Додаткові:

1. The Java Tutorials / Oracle: official tutorials [Електронний ресурс] : [Сайт]. – Режим доступу : <https://docs.oracle.com/javase/tutorial/>
2. Java Platform, Enterprise Edition: The Java EE Tutorial / Oracle: official tutorial [Електронний ресурс] : [Сайт]. – Режим доступу : <https://docs.oracle.com/javaee/7/tutorial/index.html>
3. Java Platform, Standard Edition 8 API Specification / Oracle: official documentation [Електронний ресурс] : [Сайт]. – Режим доступу : <https://docs.oracle.com/javase/8/docs/api/index.html>
4. Java Notes for Professionals / GPL: electronic book [Електронний ресурс] : [Сайт]. – Режим доступу : <https://books.goalkicker.com/JavaBook/>
5. Git Notes for Professionals / GPL: electronic book [Електронний ресурс] : [Сайт]. – Режим доступу : <https://goalkicker.com/GitBook/>
6. Hibernate Notes for Professionals / GPL: electronic book [Електронний ресурс] : [Сайт]. – Режим доступу : <https://goalkicker.com/HibernateBook/>
7. Spring Notes for Professionals / GPL: electronic book [Електронний ресурс] : [Сайт]. – Режим доступу : <https://goalkicker.com/SpringFrameworkBook/>