



## СИЛАБУС

навчальної дисципліни

Програмування на Java

(назва)

### 1. Основна інформація про дисципліну

Тип дисципліни: вибіркова      Форма контролю: залік

Освітній ступінь: бакалавр

Для спеціальності(-тей): 014.04 Середня освіта (Математика), 014.09 Середня освіта (Інформатика), 014.10 Середня освіта (Трудове навчання та технології)

Обмеження для вибору: відсутні

Кількість кредитів (годин): 4 (денна форма: 120 год.: 16 - лекції; 16 – практичні; 16-лабораторні; 72 - самостійна робота; заочна форма: 120 год.: 4 - лекції; 4 – практичні; 4-лабораторні; 108 - самостійна робота)

Мова викладання: українська

### 2. Інформація про викладача

ПІБ: Арнаут Віктор Якимович

Науковий ступінь, вчене звання, посада: викладач

Кафедра: математики, інформатики та інформаційної діяльності

Робочій e-mail: v.arnaut@idguonline.net

Години консультацій на кафедрі: середа, 15:00-16:00 (ЦНІТ)

### 3. Цілі дисципліни та результати навчання

**Предмет дисципліни:** Розробка програмного забезпечення на базі мультиплатформних технологій Java SE (Java Standart Edition), вивчення теоретичних засад та сучасних технологій проектування платформи-незалежного програмного забезпечення комп'ютерних систем збору та обробки вимірювальних даних та одержання практичних навичок їх реалізації. Особливості реалізації в Java принципів ООП та класів. Типи даних та оператори. Класи та об'єкти. Успадкування та поліморфізм. Внутрішні класи. Інтерфейси та анотації. Рядки. Графічний інтерфейс користувача. Виключення та помилки. Поток вводу-виводу. Фреймворк колекцій. Поток виконання. Патерни проектування.

**Мета дисципліни:** надання студентам необхідних знань та навичок для успішної розробки програмного забезпечення на мові Java.

**Результати навчання:** вміння ефективно використовувати можливості платформи-незалежних технологій Java для вирішення задач обробки, аналізу, збереження, відображення, передачі експериментальних даних в інформаційно-вимірювальних системах.

### 4. Зміст дисципліни

Тема № 1. **Основи Java:** Синтаксис мови Java. Змінні та типи даних. Оператори та вирази

Тема № 2. **Структури управління:** Умовні оператори (if, else, switch). Цикли (for, while, do-while). Керування виключеннями (try, catch, finally)

Тема № 3. **ООП в Java:** Класи та об'єкти. Наслідування, поліморфізм, інкапсуляція. Абстракція та інтерфейси

Тема №4. **Робота зі строками та масивами:** Рядки в Java. Робота з масивами. Колекції (ArrayList, LinkedList, HashMap)

Тема №5. **Введення/виведення даних:** Клас Scanner для введення даних. Виведення в консоль (System.out.println)

Тема №6. **Обробка файлів:** Читання та запис до текстових файлів. Робота з файловими потоками

Тема №7. **Потоки виконання (Threads):** Основні поняття про потоки. Синхронізація потоків

Тема №8. **Робота з базами даних:** Підключення до бази даних (JDBC). Виконання SQL-запитів з Java

## **5. Політика курсу**

### **Відвідування навчальних занять**

Згідно з «Положенням про порядок оцінювання рівня навчальних досягнень здобувачів вищої освіти в умовах ЄКТС» студенти мають обов'язково бути присутніми на семінарських заняттях. Студент, який з поважних причин, підтверджених документально, не мав був відсутній на семінарському занятті, має право на відпрацювання у двотижневий термін після повернення до навчання. Студент, який не використав надане йому право у встановлений термін або пропустив заняття без поважних причин, отримує за кожне пропущення заняття 0 балів. Студенти, які навчаються за індивідуальним графіком, мають в повному обсязі виконати додаткові індивідуальні завдання, попередньо узгодивши їх з викладачем. Присутність на модульній контрольній роботі є обов'язковою. У випадку відсутності студента на проміжному контролі з поважної причини, підтвердженої документально, йому призначається інша дата складання модульної контрольної роботи.

### **Академічна доброчесність**

Студенти мають дотримуватись правил академічної доброчесності відповідно до «Кодексу академічної доброчесності та корпоративної етики ІДГУ». Наявність академічного плагіату в студентських роботах є підставою для виставлення негативної оцінки. Списування студентів під час проведення модульної контрольної роботи є підставою для дострокового припинення її складання та виставлення негативної оцінки.

### **Використання технологій штучного інтелекту**

Навчальна дисципліна не передбачає використання здобувачами генеративних моделей штучного інтелекту. Генерація тексту для виконання індивідуальних занять або підготовки до семінарських занять буде кваліфікуватися як порушення академічної доброчесності.

## **6. Контрольні заходи та критерії оцінювання**

### **Шкала та схема формування підсумкової оцінки**

| СУМА БАЛІВ | ОЦІНКА ЗА НАЦІОНАЛЬНОЮ ШКАЛОЮ |
|------------|-------------------------------|
| 90-100     | зараховано                    |
| 70-89      |                               |
| 51-69      |                               |
| 1-50       | незараховано                  |

Підсумковий бал з навчальної дисципліни виставляється за результатами поточного, проміжного та підсумкового контролю. Під час поточного контролю оцінюються відповіді студента на семінарських заняттях та результати самостійної роботи. Нарахування балів за поточний контроль відбувається відповідно до «Положення про порядок оцінювання рівня навчальних досягнень здобувачів вищої освіти в умовах ЄКТС в ІДГУ» [http://idgu.edu.ua/wpcontent/uploads/2020/09/polozhennja\\_pro\\_porjadok\\_ocinjuvannja\\_rivnja\\_navchalnyh\\_dosjahrenn\\_z\\_i\\_zminamy-vid-28.08.2020-protokol-1.pdf](http://idgu.edu.ua/wpcontent/uploads/2020/09/polozhennja_pro_porjadok_ocinjuvannja_rivnja_navchalnyh_dosjahrenn_z_i_zminamy-vid-28.08.2020-protokol-1.pdf).

### **Форма проміжного контролю**

Результати вивчення навчальної дисципліни оцінюються за допомогою модульної контрольної роботи.

## **Форма підсумкового контролю**

Підсумковий контроль - залік за результатами поточного та проміжного контролю.

### **Критерії оцінювання під час аудиторних занять**

- Здатність пояснити основні поняття Java, такі як синтаксис, ООП, обробка винятків і т. д.
- Розуміння основних структур даних та алгоритмів.
- Здатність писати код на Java відповідно до поставленого завдання.
- Ефективне використання мови та стандартних бібліотек.
- Вміння використовувати різні типи даних та їх перетворення.
- Коректне використання змінних та констант.
- Здатність створювати класи та використовувати принципи ООП.
- Наслідування, поліморфізм та інкапсуляція в практиці.
- Вміння працювати з масивами та використовувати колекції для обробки даних.
- Здатність вирішувати завдання, пов'язані із списками, масивами, чергами тощо.
- Вміння взаємодіяти з користувачем через консоль.
- Здатність читати та записувати дані в файли.
- Навички написання та виконання SQL-запитів.
- Здатність розробляти функціональні програми для конкретних завдань.
- Ефективність та правильність рішення.
- Активна участь у дискусіях та вирішенні завдань під час занять.

### **Критерії оцінювання завдань самостійної роботи**

- Забезпечення правильності розв'язання завдань.
- Ефективне використання мови Java та її можливостей.
- Чітка структура та іменування змінних та методів.
- Дотримання принципів чистого коду.
- Застосування наслідування, поліморфізму та інкапсуляції.
- Створення класів та об'єктів за необхідністю.
- Наявність коментарів та пояснень до коду.
- Документування коду для зрозумілості іншим програмістам.
- Здатність виявляти та вирішувати проблеми в коді.
- Ефективне використання доступних ресурсів для допомоги (документація, форуми, тощо).
- Вчасне та систематичне виконання завдань.
- Дотримання визначених строків та обсягів роботи.

### **Критерії оцінювання модульної контрольної роботи**

- Коректність та точність виконання завдань.
- Відсутність логічних та синтаксичних помилок у коді.
- Відповідність принципам чистого коду та іменування змінних та методів.
- Ефективне використання можливостей мови Java.
- Мінімізація зайвих операцій та оптимізація швидкодії коду.
- Наявність тестів для перевірки роботи програми.
- Обробка винятків та ситуацій, що можуть призвести до помилок.
- Наявність коментарів та пояснень до коду.
- Зрозумілість та легкість розуміння коду іншими програмістами.
- Коректне використання інтегрованих середовищ розробки (IDE).
- Використання додаткових інструментів, які полегшують розробку (наприклад, системи контролю версій).
- Використання творчого підходу до завдань.
- Додаткові функціональності, які покращують роботу програми.
- Виконання завдань у визначений термін.
- Відповідність вимогам та обсягу роботи, зазначеному в завданні.

### **Критерії оцінювання під час підсумкового контролю**

- Здатність студента визначати ключові концепції та терміни, пов'язані з Java.
- Розуміння базових і продвинутих аспектів мови програмування.
- Вміння студента застосовувати вивчені концепції та навички у вирішенні завдань.
- Здатність писати чистий та ефективний код на Java.
- Знання та використання принципів об'єктно-орієнтованого програмування (ООП).
- Здатність відновлювати та обробляти помилки для покращення надійності програм.
- Використання різних типів даних та структур даних в Java.
- Чітка структура програм, відділення функціональності, дотримання принципів чистого коду.
- Організованість та систематичність в роботі.
- Знання та використання тестування для перевірки правильності та працездатності коду.
- Наявність тестових випадків для ключових функціональностей.
- Чіткі та інформативні коментарі до коду.
- Наявність опису важливих частин програми, документаційних коментарів.
- Творчий підхід, застосування додаткових функцій чи покращень.
- Дотримання завдання, відповідність вимогам.

### **7. Основна література та інформаційні ресурси**

1. Bruce Eckel. Thinking in Java. Prentice Hall, 2006. 1150 p.
2. Herbert Schildt. Java: The Complete Reference, Twelfth Edition 12th Edition. McGraw Hill, 2022. 1280 p.
3. Java-програмування: комп'ютерний практикум [Електронний ресурс] : навч. посіб. для студ. спеціальності 122 «Комп'ютерні науки», освітньо-професійної програми «Комп'ютерний моніторинг та геометричне моделювання процесів і систем» / КПІ ім. Ігоря Сікорського ; уклад.: Ю. А. Тарнавський. Київ : КПІ ім. Ігоря Сікорського, 2021. 95 с.
4. Васильєв О.М. Програмування мовою Java. Тернопіль: Навчальна книга – Богдан, 2020 р. 696 с.
5. Ерік Фрімен, Елізабет Робсон. Head First. Патерни проектування. Фабула, 2020. 688с.
6. Ерік Фрімен (2020). Head First. Патерни проектування. Легкий для сприйняття довідник.
7. Кетті Сьєрра, Берт Бейтс. Head First. Java. Фабула, 2022. 720 с.
8. Роберт Мартін. Чиста архітектура. Ранок, 2019. 368 с.
9. Роберт Мартін. Чистий код. Створення і рефакторинг за допомогою Agile. Фабула, 2019. 448 с.

Затверджено на засіданні кафедри математики, інформатики та інформаційної діяльності (протокол № 8 від 16 січня 2024 р.)