Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра ЕОМ



Звіт

до лабораторної роботи №1

з дисципліни: “Кросплатформні засоби програмування”

на тему: «Дослідження базових конструкцій мови Java»

Варіант №12

Виконала:

ст.гр. КІ-307

Коваль К. О.

Прийняв:

Іванов Ю. С.

**Львів 2023**

**Мета:** ознайомитися з базовими конструкціями мови Java та оволодіти навиками написання й автоматичного документування простих консольних програм мовою Java.

**Завдання:**

1. Написати та налагодити програму на мові Java згідно варіанту. Програма має задовольняти наступним вимогам:

• програма має розміщуватися в загальнодоступному класі Lab1ПрізвищеГрупа;

• програма має генерувати зубчатий масив, який міститиме лише заштриховані області квадратної матриці згідно варіанту;

• розмір квадратної матриці і символ-заповнювач масиву вводяться з клавіатури;

• при не введені або введенні кількох символів-заповнювачів відбувається коректне переривання роботи програми;

• сформований масив вивести на екран і у текстовий файл;

• програма має володіти коментарями, які дозволять автоматично згенерувати документацію до розробленої програми.

2. Автоматично згенерувати документацію до розробленої програми.

3. Завантажити код на GitHub згідно методичних вказівок по роботі з GitHub.

4. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації та завантажити його у ВНС.

5. Дати відповідь на контрольні запитання:  
1. Які дескриптори використовуються при коментуванні класів?

2. Які дескриптори використовуються при коментуванні методів?

3. Як автоматично згенерувати документацію?

4. Які прості типи даних підтримує Java?

5. Як оголосити змінну-масив?

6. Які керуючі конструкції підтримує Java?

7. В чому різниця між різними варіантами оператора for?

8. Як здійснити ввід з консолі?

9. Як здійснити ввід з текстового файлу?

10. Як здійснити запис у текстовий файл?

**Варіант №12:**



**Виконання:**

import java.io.\*;

import java.util.\*;

/\*\*

\* Клас Lab1KovalKI307 реалізує програму для створення зубчатого масиву

\* і запису його у текстовий файл та виведення на екран.

\*/

public class Lab1KovalKI307 {

/\*\*

\* Головний метод, що виконує основну логіку програми.

\*

\* @param args Аргументи командного рядка (не використовуються в цій програмі).

\* @throws FileNotFoundException Виникає, якщо файл не може бути створений або відкритий для запису.

\*/

public static void main(String[] args) throws FileNotFoundException {

int nRows; // Змінна для зберігання розміру матриці

char[][] arr; // Матриця для зберігання символів

String filler; // Символ-заповнювач

Scanner in = new Scanner(System.in); // Об'єкт для отримання вводу з клавіатури

File dataFile = new File("MyFile.txt"); // Файл для запису результатів

PrintWriter fout = new PrintWriter(dataFile); // Об'єкт для запису до файлу

// Просимо користувача ввести розмір квадратної матриці

System.out.print("Введіть розмір квадратної матриці: ");

nRows = in.nextInt();

in.nextLine(); // Перехід на наступний рядок після введення числа

// Просимо користувача ввести символ-заповнювач

System.out.print("Введіть символ-заповнювач: ");

filler = in.nextLine();

// Перевірка на правильність введення символу-заповнювача

if (filler.length() != 1) {

System.out.println("Помилка: Символ-заповнювач повинен бути рівно одним символом.");

return; // Виходимо з програми

}

// Ініціалізація матриці розміром nRows x nRows

arr = new char[nRows][nRows];

// Заповнюємо ліву верхню половину і праву нижню половину матриці символом

for (int i = 0; i < nRows; i++) {

for (int j = 0; j < nRows; j++) {

if (i < nRows / 2 && j < nRows / 2) {

arr[i][j] = filler.charAt(0);

} else if (i >= nRows / 2 && j >= nRows / 2) {

arr[i][j] = filler.charAt(0);

} else {

arr[i][j] = ' ';

}

fout.print(arr[i][j] + " "); // Записуємо символ у файл

}

fout.println(); // Перехід на наступний рядок у файлі

}

// Виводимо матрицю на екран

for (int i = 0; i < nRows; i++) {

for (int j = 0; j < nRows; j++) {

System.out.print(arr[i][j] + " ");

}

System.out.println(); // Перехід на наступний рядок у консолі

}

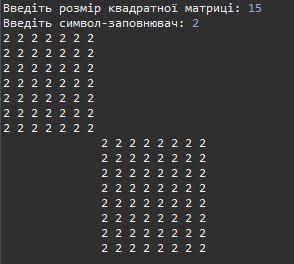
fout.flush(); // Зберігаємо дані у файл

fout.close(); // Закриваємо файл

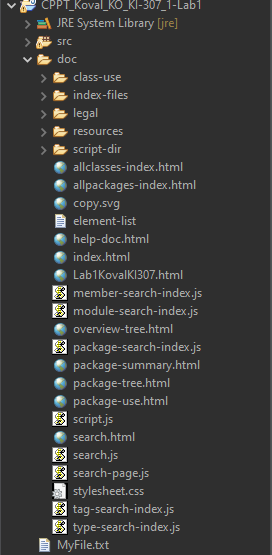
}

}

**Результат:**

****

**Документація:**

****

**Відповіді на контрольні запитання:**

1. \*\*Дескриптори для коментування класів в Java:\*\*

- `@class`: Опис класу.

2. \*\*Дескриптори для коментування методів в Java:\*\*

- `@param`: Опис параметрів методу.

- `@return`: Опис значення, яке повертає метод.

- `@throws` або `@exception`: Опис винятків, які може викинути метод.

- `@deprecated`: Помічає метод як застарілий.

3. \*\*Автоматичне генерування документації в Java:\*\*

- Використання утиліти Javadoc: Додається коментар `/\*\* ... \*/` перед класом або методом, і потім викликається Javadoc для створення документації.

4. \*\*Прості типи даних в Java:\*\*

- Цілі числа: `byte`, `short`, `int`, `long`

- Дійсні числа: `float`, `double`

- Символи: `char`

- Логічний тип: `boolean`

5. \*\*Оголошення змінної-масиву в Java:\*\*

- `тип\_даних[] ім'я\_змінної = new тип\_даних[розмір];`

6. \*\*Керуючі конструкції в Java:\*\*

- `if`, `else`

- `switch`

- `for`, `while`, `do-while`

- `break`, `continue`

- `return`

7. \*\*Різниця між різними варіантами оператора for в Java:\*\*

- `for(initialization; condition; update)`: Базовий варіант.

- `for(:)`: Enhanced for loop для ітерації по колекціях.

- `for( : : )`: Введений в Java 8 для ітерації по стрімах у функціональному стилі.

8. \*\*Ввід з консолі в Java:\*\*

- Використання `Scanner` класу або `BufferedReader` класу для зчитування введених даних з консолі.

9. \*\*Ввід з текстового файлу в Java:\*\*

- Використання `Scanner` або `BufferedReader` для читання даних з файлу.

10. \*\*Запис у текстовий файл в Java:\*\*

- Використання `FileWriter` або `BufferedWriter` для запису даних у файл.

**Висновок:** на даній лабораторній роботі я ознайомилась з базовими конструкціями мови Java та оволоділа навиками написання й автоматичного документування простих консольних програм мовою Java.