

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Кафедра «Електронних обчислювальних машин»



Звіт
з лабораторної роботи № 1
з дисципліни: «Кросплатформенні засоби програмування»
на тему: «Дослідження базових конструкцій мови Java»

Виконав:

студент групи КІ-306

Гапонова Дарина

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

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

Завдання (варіант № 4)

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 введенні 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.

Вихідний код програми

```
import java.io.*;
import java.util.*;
/**
 * The Lab1 class implements an example program for laboratory work №1
 *
 * @author No Information
 * @version 1.0
 * @since version 1.0
 */
public class lab1
{
    public static void main(String[] args){
        int size = inputSize();
```

```

        char symbol = inputFiller();
        char[][] jaggedArr = generateJaggedArr(size, symbol);

        showJaggedArr(jaggedArr);
        saveJaggedArrToFile(jaggedArr);
    }

    public static int inputSize() {
        Scanner scanner = new Scanner(System.in);
        int size;

        while(true) {
            try {
                System.out.print("Enter size of square matrix: ");
                size = scanner.nextInt();
                break;
            }
            catch (Exception e) {
                System.out.println(e.getMessage() + ". Please enter a number");
                scanner.next();
            }
        }
        return size;
    }

    public static char inputFiller() {
        Scanner scanner = new Scanner(System.in);
        char symbol;

        while(true) {
            try {
                System.out.print("Enter character to fill matrix: ");
                String str = scanner.nextLine();

                if (str.isEmpty()) {
                    continue;
                }
                symbol = str.charAt(0);
                break;
            }
            catch (Exception e) {
                System.out.println(e.getMessage() + ". Please enter a
character");
                scanner.next();
            }
        }
        return symbol;
    }

    public static char[][] generateJaggedArr(int size, char symbol) {
        char[][] jaggedArr = new char[size][];

        int endOfArrPointer = 0;
        int center = size / 2;

        for (int i = 0; i < center; i++) {
            endOfArrPointer++;
            jaggedArr[i] = new char[endOfArrPointer];
            Arrays.fill(jaggedArr[i], ' ');
        }

        for (int i = center; i < size; i++) {
            endOfArrPointer++;
            jaggedArr[i] = new char[endOfArrPointer];
            Arrays.fill(jaggedArr[i], ' ');
        }
    }

```

```

        for (int j = size - i - 1; j <= i; j++) {
            jaggedArr[i][j] = symbol;
        }
    }

    return jaggedArr;
}

public static void showJuggedArr(char[][] jaggedArr) {
    for (char[] charArr : jaggedArr) {
        for (char ch : charArr) {
            System.out.print(ch + " ");
        }
        System.out.println();
    }
}

public static void saveJuggedArrToFile(char[][] jaggedArr) {
    try {
        PrintWriter writer = new PrintWriter("lab1/MyFile.txt");

        for (char[] charArr : jaggedArr) {
            for (char ch : charArr) {
                writer.print(ch + " ");
            }
            writer.print("\n");
        }

        writer.flush();
        writer.close();
    }
    catch (FileNotFoundException e) {
        System.out.println(e.getMessage());
    }
}
}

```

Результат виконання програми

```

Enter size of square matrix: 9
Enter character to fill matrix: 0

```

```

      0
    0 0 0
  0 0 0 0 0 |
0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0

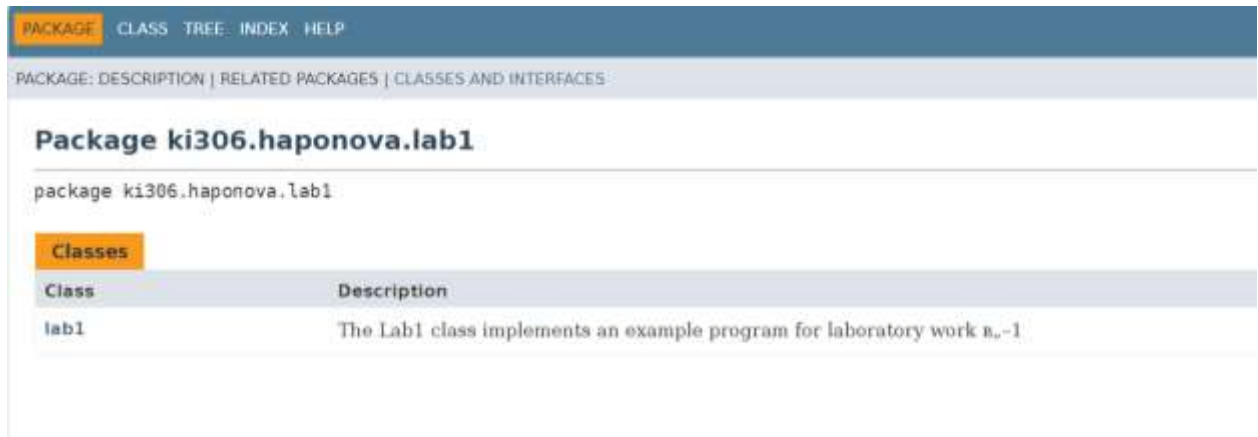
```

```

Process finished with exit code 0

```

Фрагмент згенерованої документації



Відповіді на контрольні запитання

1. Які дескриптори використовуються при коментуванні класів?

```
/**
```

Об'єкт класу Person описує особу.

Особа має властивості: ім'я, прізвище та стать.

```
*/
```

2. Які дескриптори використовуються при коментуванні методів?

- @param змінна опис

Цей дескриптор додає в опис методу розділ “parameters”. Опис цього елементу може складатися з кількох рядків та містити html-теги. Всі дескриптори @param, що відносяться до одного методу слід групувати разом.

- @return опис

Цей дескриптор додає в опис методу розділ “returns”. Опис цього елементу може складатися з кількох рядків та містити html-теги.

- @throws опис_класу

Цей дескриптор додає в опис методу інформацію про класи об'єкти яких можуть генеруватися при виключних ситуаціях. Відомості про кожен клас слід описувати в окремому дескрипторі @throws.

3. Як автоматично згенерувати документацію?

Для генерування документації по пакету слід ввести в консолі ОС Windows:

```
javadoc -d каталог_doc ім'я_пакету
```

4. Які прості типи даних підтримує Java?

boolean, char, byte, short, int, long, float, double.

5. Як оголосити змінну-масив?

```
тип[] змінна;
```

```
тип змінна[];
```

6. Які керуючі конструкції підтримує Java?

Такі оператори як switch, if-else, while, do-while – ідентичні аналогічним конструкціям у мовах C/C++. Оператор циклу for має деякі особливості. У Java цей оператор має 2 різновиди:

- конструкція в стилі C/C++ з полем ініціалізації, логічною умовою та кроком;
- конструкція з синтаксисом foreach.

7. В чому різниця між різними варіантами оператора for?

Різниця полягає в застосуванні лічильника.

8. Як здійснити ввід з консолі?

Для введення інформації з консолі необхідно створити об'єкт класу Scanner і зв'язати його з стандартним потоком вводу System.in, наприклад:

```
Scanner in = new Scanner(System.in);
```

9. Як здійснити ввід з текстового файлу?

Для введення інформації з файлу необхідно підключити пакет java.io та створити об'єкт класу Scanner з об'єкту File:

```
Scanner fin = new Scanner(File("MyFile.txt"));
```

10. Як здійснити запис у текстовий файл?

Для виведення інформації у текстовому вигляді у файл треба підключити пакет java.io та створити об'єкт класу PrintWriter в конструкторі якого необхідно вказати назву файлу, що відкривається на запис, наприклад: `PrintWriter fout = new PrintWriter("MyFile.txt");`

Висновок

У ході виконання даної лабораторної роботи, я ознайомилась з базовими конструкціями мови Java та оволоділа навиками написання й автоматичного документування простих консольних програм мовою Java.