

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
Бокало П.М.

Прийняв:

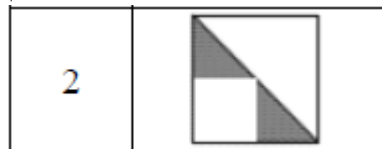
доцент кафедри ЕОМ
Іванов Ю. С.

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

Завдання(варіант №2):

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.

Індивідуальне завдання:



Вихідний код програми:

```
import java.io.*;
import java.util.*;
/**
 * Ця програма виводить в консоль та у файл рисунок згідно до 2 варіанту
 * @author Petro Bokalo KI-203
 * @version 1.0
 * @since version 1.0
 */

public class Lab1BokaloKI306 {

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

```

String symbol;
int N;
char[][] matrix;
PrintWriter fout = new PrintWriter("Lab1.txt");

Scanner input = new Scanner(System.in);
System.out.print("Input number of elements : ");
N = input.nextInt();
System.out.print("Your matrix will have " + N + " rows and columns\n");

if (N < -1) {
    N *= -1;
} else if (N == 0) {
    System.out.print("Invalid number for N");
    System.exit(0);
} else if (N % 2 != 0) {
    N += 1;
    System.out.print("Your number was odd, so your number is now : " + N +
"\n");
}

System.out.print("\nInput your placeholder symbol : ");
input.nextLine();

symbol = input.nextLine();
System.out.print("\nYour placeholder is : " + symbol + "\n");

if (symbol.length() > 1) {
    System.out.print("You've entered too many symbols");
    System.exit(0);
}

// Ініціалізуємо зубчастий масив потрібною кількістю рядків
int checker = 0;

matrix = new char[N][];

for (int i = 0; i < N ; i++) {
    if(checker < N/2)
    {
        matrix[i] = new char[i + 1];
        for (int j = 0; j <= i; j++) {
            matrix[i][j] = symbol.charAt(0);
            System.out.print(matrix[i][j] + "\t");
            fout.write(matrix[i][j] + "\t");
        }
    }
}

```

```

    }
    System.out.println();
    fout.write("\n");
    }

else
{
    matrix[i] = new char[i + 1];

    for(int j = N/2; j > 0; j--)
    {

        System.out.print("\t");
        fout.write("\t");

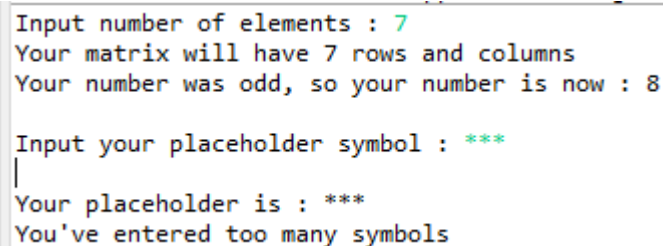
    }

    for (int j = N/2; j <= i; j++) {
        matrix[i][j] = symbol.charAt(0);
        System.out.print(matrix[i][j] + "\t");
        fout.write(matrix[i][j] + "\t");
    }
    System.out.println();
    fout.write("\n");
}

checker++;
}
fout.close();
input.close();
System.out.close();
}
}

```

Результати роботи програми:



```

Input number of elements : 7
Your matrix will have 7 rows and columns
Your number was odd, so your number is now : 8

Input your placeholder symbol : ***
|
Your placeholder is : ***
You've entered too many symbols

```

Рисунок 1. Випадок коли введено забагато символів

```

Input number of elements : 8
Your matrix will have 8 rows and columns

Input your placeholder symbol : *
|
Your placeholder is : *
*
*      *
*      *      *
*      *      *      *
                                *
                                *      *
                                *      *      *
                                *      *      *      *

```

Рисунок 2. Випадок коректного введення даних

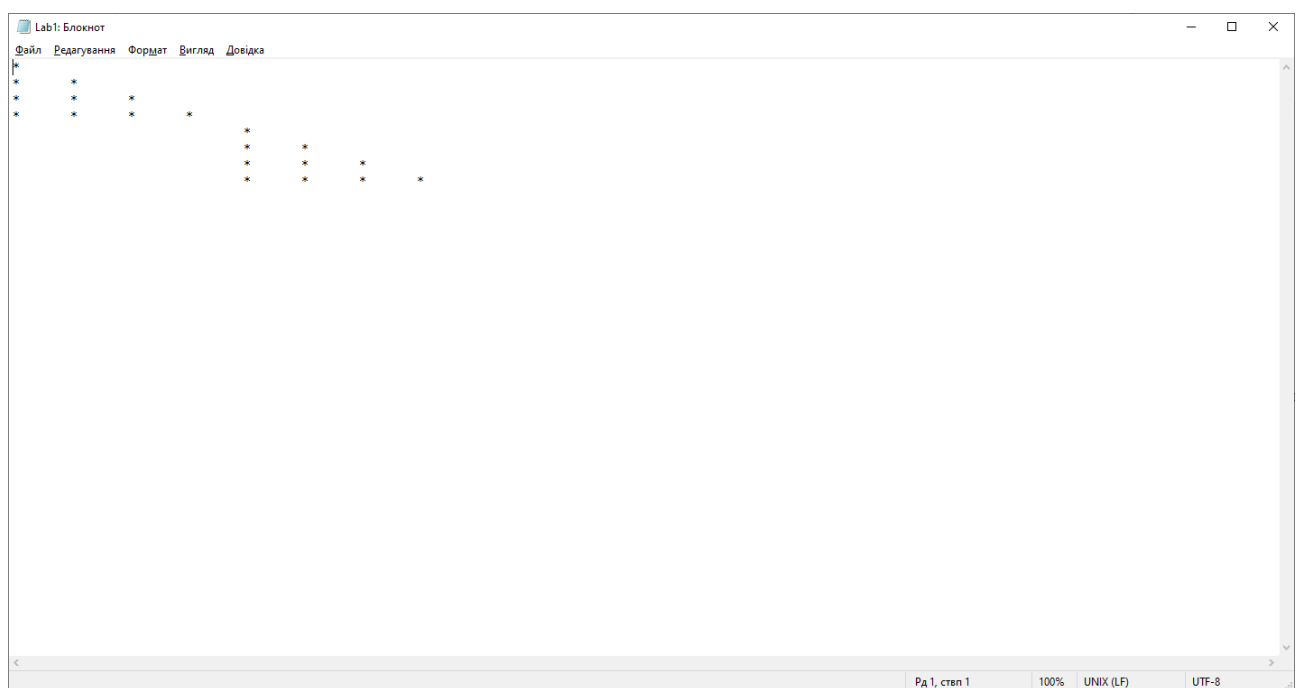


Рисунок 3. Файл після виконання програми

Фрагменти згенерованої документації:

All Classes and Interfaces

Classes	
Class	Description
Lab1BokaloKI306	Ця програма виводить в консоль та у файл рисунок згідно до 2 варіанту

Рисунок 1. Фрагмент документації

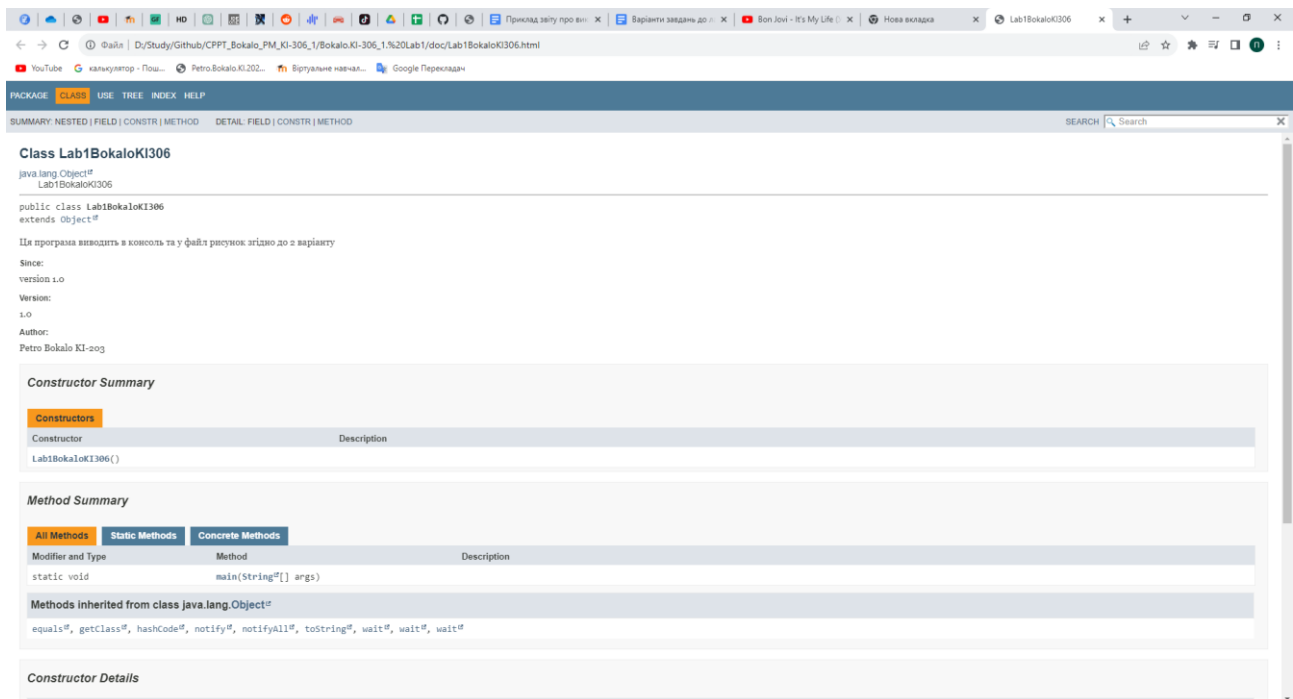


Рисунок 2. Фрагмент документації

Відповідь на контрольні питання:

1. Які дескриптори використовуються при коментуванні класів?
*Коментарі, що застосовуються до класів, використовують дескриптор `/** ... */`.*
2. Які дескриптори використовуються при коментуванні методів?
*Для документування методів використовують дескриптор `/** ... */`.*
3. Як автоматично згенерувати документацію?
Щоб автоматично згенерувати документацію, необхідно вибрати "Project -> Generate Javadoc".
4. Які прості типи даних підтримує Java?
Boolean, char, byte, short, int, long, float, double.
5. Як оголосити змінну-масив?
Для створення масиву цілих чисел використовується синтаксис: `int[] myArray;`
6. Які керуючі конструкції підтримує Java?
Java підтримує: if, else, for, while, do-while, switch, break, continue, return.
7. В чому різниця між різними варіантами оператора for?
Оператор циклу for з синтаксисом foreach використовується для ітерації по елементах масиву. Синтаксис: for (змінна : набір даних), де "змінна" представляє поточний елемент набору даних
8. Як здійснити ввід з консолі?

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

9. Як здійснити ввід з текстового файлу?

Для введення інформації з файлу необхідно підключити пакет `java.io` і створити об'єкт класу `Scanner`, який приймає об'єкт типу `File` як параметр конструктора.

10. Як здійснити запис у текстовий файл?

Для виведення інформації у текстовому вигляді у файл, необхідно підключити пакет `java.io` і створити об'єкт класу `PrintWriter`, який приймає назву файлу, що відкривається на запис, як параметр конструктора.

Висновок: На цій лабораторній роботі я ознайомився із мовою програмування Java. Застосував зубчасті масиви. Згенерував документацію. Виклав проект на Github і реалізував вивід рисунку на консоль і у файл.