

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

Довганюк О.С.

**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 № 6)

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

- програма має розміщуватися в загальнодоступному класі Lab2ПрізвищеГрупа;
- програма має генерувати зубчатий масив, який міститиме лише заштриховані області квадратної матриці згідно варіанту (рис. 1);

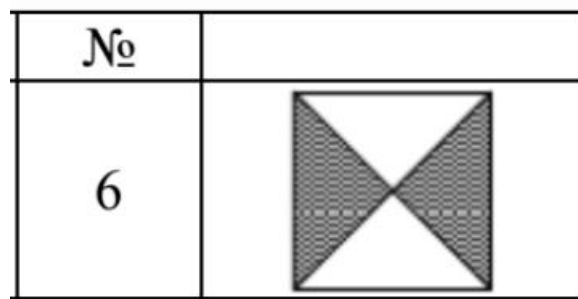


Рис. 1. Заштрихована область квадратної матриці.

- 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. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.
4. Дати відповіді на контрольні запитання:
- які дескриптори використовуються при коментуванні класів?
  - які дескриптори використовуються при коментуванні методів?
  - як автоматично згенерувати документацію?

- які прості типи даних підтримує java?
- як оголосити змінну-масив?
- які керуючі конструкції підтримує java?
- в чому різниця між різними варіантами оператора for?
- як здійснити ввід з консолі?
- як здійснити ввід з текстового файлу?
- як здійснити запис у текстовий файл?

## Вихідний код програми

```
import java.io.*;
import java.util.*;

/**
 * The Lab1DovganiukKI306 class implements a program for generating and inferring a jagged array of characters.
 *
 * @author Donganiuk Olexandr
 * @version 1.0
 * @since version 1.0
 */
public class Lab1DovganiukKI306 {
    /**
     * The static main method is the entry point into the program
     *
     * @param args
     * @throws FileNotFoundException
     */
    public static void main(String[] args) throws FileNotFoundException {
        int nRows;
        char[][] arr;
        Scanner in = new Scanner(System.in);
        File dataFile = new File("MyFile.txt");
        PrintWriter fout = new PrintWriter(dataFile);
        System.out.print("Введіть розмір квадратної матриці: ");
        nRows = in.nextInt();
        in.nextLine();
    }
}
```

```

arr = new char[nRows][];

/*
for (int i = 0; i < nRows; i++) {
    if (i < nRows/2) {
        arr[i] = new char[2 * (i + 1)];
    }
    else if (2 * (i + 1) > nRows + 2) {
        arr[i] = new char[2 * (nRows - i)];
    }
    else {
        arr[i] = new char[nRows];
    }
}
*/

for (int i = 0; i < nRows; i++) {
    arr[i] = i < nRows / 2 ?
        new char[2 * (i + 1)] :
        2 * (i + 1) > nRows + 2 ?
            new char[2 * (nRows - i)] :
            new char[nRows];
}

System.out.print("Введіть символ-заповнювач: ");
String filler = in.nextLine();

if (filler.isEmpty()) {
    System.out.print("\nНе введено символ заповнювач");
    return;
}
else if (filler.length() != 1) {
    System.out.print("\nЗабагато символів заповнювачів");
    return;
}

System.out.println("\nЗубчатий масив:");

```

```

for (int i = 0; i < nRows; i++) {
    for (int j = 0; j < arr[i].length; j++) {
        arr[i][j] = filler.charAt(0);
        System.out.print(arr[i][j] + " ");
    }
    System.out.println();
}

```

```

System.out.println("\nЗадана матрица:");

```

```

for (int i = 0; i < nRows; i++) {
    int k = 0;
    for (int j = 0; j < nRows; j++) {
        if (i == j || j == nRows - 1 - i || (i + j < nRows && i - j > 0) || (i + j + 1 > nRows && i - j < 0)) {
            System.out.print(arr[i][k] + " ");
            fout.print(arr[i][k] + " ");
            k++;
        }
        else {
            System.out.print(" ");
            fout.print(" ");
        }
    }
    System.out.println();
    fout.println();
}

```

```

fout.flush();

```

```

fout.close();

```

```

}
}

```

## Результат виконання програми

Введіть розмір квадратної матриці: 9

Введіть символ-заповнювач: +

Зубчатий масив:

+ +

+ + + +

+ + + + + +

+ + + + + + + +

+ + + + + + + + +

+ + + + + + + +

+ + + + + +

+ + + +

+ +

Задана матриця:

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

+ + + + + + + +

Process finished with exit code 0

## Текстовий файл з результатом виконання програми

MyFile.txt ×	
1	+ + + + + + + +
2	+ + + + + + + +
3	+ + + + + + + +
4	+ + + + + + + +
5	+ + + + + + + +
6	+ + + + + + + +
7	+ + + + + + + +
8	+ + + + + + + +
9	+ + + + + + + +
10	+ + + + + + + +

## Фрагмент згенерованої документації

PACKAGE

CLASS

USE

TREE

INDEX

HELP

SUMMARY: NESTED | FIELD | CONSTR | METHODDETAIL: FIELD | CONSTR | METHOD

### Class Lab1DovganiukKI306

java.lang.Object<sup>↗</sup>  
Lab1DovganiukKI306

```
public class Lab1DovganiukKI306
extends Object↗
```

The Lab1DovganiukKI306 class implements a program for generating and inferring a jagged array of characters.

**Since:**  
version 1.0

**Version:**  
1.0

**Author:**  
Donganiuk Olexandr

#### Constructor Summary

Constructors	
Constructor	Description
Lab1DovganiukKI306()	

#### Method Summary

All Methods	Static Methods	Concrete Methods
Modifier and Type	Method	Description
static void	main(String <sup>↗</sup> [] args)	The static main method is the entry point into the program

## Відповіді на контрольні запитання

### 1. Які дескриптори використовуються при коментуванні класів?

Дескриптори, які використовуються при коментуванні класів в Java, включають: `@author`, `@version`, `@since`, `@deprecated`, `@see`.

### 2. Які дескриптори використовуються при коментуванні методів?

Дескриптори, які використовуються при коментуванні класів в Java, включають: `@param`, `@throws`, `@return`.

### 3. Як автоматично згенерувати документацію?

Документацію можна автоматично згенерувати за допомогою інструменту Javadoc, додавши коментарі `/** ... */` перед класами, методами та полями, та використовуючи команду `javadoc` у командному рядку.

### 4. Які прості типи даних підтримує Java?

Java підтримує прості типи даних, такі як цілі числа (`int`, `short`, `long`), дійсні числа (`float`, `double`), символи (`char`), логічні значення (`boolean`) і байти (`byte`).

## **5. Як оголосити змінну-масив?**

Змінну-масив оголошують, вказавши тип даних, після чого вказують квадратні дужки, наприклад: `int[] arr`.

## **6. Які керуючі конструкції підтримує Java?**

Java підтримує керуючі конструкції, такі як `if-else`, `switch`, `for`, `while`, `do-while` для управління потоком програми.

## **7. В чому різниця між різними варіантами оператора `for`?**

Різниця між різними варіантами оператора `for` полягає в синтаксисі і області застосування. Наприклад, `for (int i = 0; i < 10; i++)` - це цикл з лічильником, а `for` (тип елемент: колекція) - це цикл `for-each` для перебору колекцій.

## **8. Як здійснити ввід з консолі?**

Ввід з консолі можна здійснити за допомогою класу `Scanner`, зчитуючи дані зі стандартного вводу (`System.in`).

## **9. Як здійснити ввід з текстового файлу?**

Для вводу з текстового файлу використовується клас `FileReader` або `BufferedReader` з `FileReader` для ефективного читання.

## **10. Як здійснити запис у текстовий файл?**

Запис у текстовий файл можна виконати за допомогою класу `FileWriter` або `BufferedWriter` з `FileWriter` для ефективного запису.

## **Висновок**

Під час лабораторної роботи, я ознайомився з базовими конструкціями мови Java та оволодів навичками написання й автоматичного документування простих консольних програм мовою Java.