

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



Звіт
з лабораторної роботи № 2
з дисципліни: «Кросплатформенні засоби програмування»
на тему: «Дослідження базових конструкцій мови Java»

Виконав:

студент групи КІ-35

Андрусяк М.В.

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 № 7)

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);

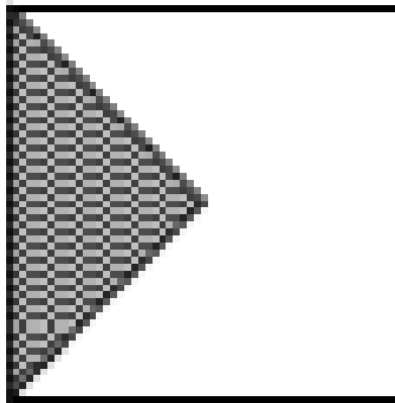


Рис. 1. Заштрихована область квадратної матриці.

- 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?

Текст програми Lab2AndrusiakKI35.java

```
package com.labs;
import java.io.*;
import java.util.*;

/*
 * Клас Lab2 реалізує програму до лабораторної роботи №2
 *
 * @author Marko Andrusiak
 * @version 1.0
 * @since version 1.0
 */

public class Lab2AndrusiakKI35 {
    /*
     * Статичний метод main є точкою входу в програму
     *
     * @param args
     * @throws FileNotFoundException
     */
}
```

```

public static void main(String[] args) throws FileNotFoundException
{
    int nRows;
    char[][] arr;
    String filler;

    Scanner in = new Scanner(System.in);

    System.out.print("Enter the size of the gear matrix, not less than 3: ");
    nRows = in.nextInt();
    if( nRows < 3 ) {
        Lab2AndrusiakKI35.main(args);
        return;
    }
    in.nextLine();

    File dataFile = new File("MyFile.txt");
    PrintWriter fout = new PrintWriter(dataFile);

    arr = new char[nRows][];

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

    System.out.print("\nEnter a placeholder character: ");
    filler = in.nextLine();

    if(filler.length() == 0) {
        System.out.println("No character entered");
    } else if(filler.length() == 1) {

```

```

        for(int i=0; i<nRows; i++) {
            for(int j=0; j<arr[i].length; j++) {
                arr[i][j]=(char)filler.codePointAt(0);
                System.out.print(arr[i][j]);
                fout.print(arr[i][j]);
            }

            fout.print("\n");
            System.out.print("\n");
        }
    } else {
        System.out.println("Too many character entered");
    }

    fout.flush();
    fout.close();

}
}

```

Результат виконання програми

```

Lab2AndrusiakKI35 x
D:\Programs\Java\bin\java.exe "-javaagent:D:\Programs\Intelli
Enter the size of the gear matrix, not less than 3: 6

Enter a placeholder character: #
#
##
###
###
##
#

Process finished with exit code 0

```

Відповіді на контрольні запитання

- Дескриптори, які використовуються при коментуванні класів: `@author`, `@version`, `@since`, `@deprecated`, `@see`.
- Дескриптори, які використовуються при коментуванні методів: `@param`, `@return`, `@throws`.
- Для того, щоб автоматично згенерувати документацію можна скористатися одним із двох методів:

1. В командному рядку ввести: `javados -d каталог_doc ім'я_паketу`

2. Скористатися відповідним інструментом, який входить в IDE.

- Java підтримує 8 простих типів даних, а саме: `boolean`, `char`, `byte`, `short`, `int`, `long`, `float`, `double`.
- Для оголошення змінної-масива потрібно слідувати за відповідною конструкторською:

тип [] змінна

- Серед керуючих інструкцій, які підтримує Java є: `switch`, `if-else`, `while`, `do-while`, `for`, `foreach`.
- Для введення інформації з консолі необхідно створити об'єкт класу `Scanner` і зв'язати його з стандартним потоком вводу `System.in`: **`Scanner in = new Scanner(System.in);`** Зробивши це ми отримуємо доступ до методів класу `Scanner`, які призначені для введення даних простих типів і рядків: `nextByte`, `nextInt` і т.ін.
- 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"));`** На цім, пошук файлу відбувається у директорії з якої була запущена на виконання програма.
- 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");`

Зробивши це ми отримуємо доступ до методів класу **`PrintWriter`**, які призначені для виведення даних простих типів і рядків: **`print`**, **`write`**

Висновок : виконуючи дану лабораторну роботу, я ознайомився з базовими конструкціями мови Java.