

Міністерство освіти і науки України

Національний університет “Львівська політехніка”

Кафедра ЕОМ



Звіт

З лабораторної роботи №1

Варіант – 3

З дисципліни: «Кросплатформні засоби програмування»

На тему: «Дослідження базових конструкцій мови JAVA»

Виконав: ст. гр. КІ-306

Братівник Д. А.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

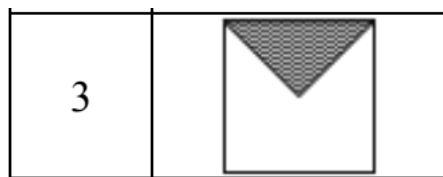
Львів 2023

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

Завдання(Варіант 3)

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; 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 відповідь на контрольні запитання:
 1. Як автоматично згенерувати документацію?
 2. В чому різниця між різними варіантами оператора for?

Варіант завдання:



Код програми:

```
/**
 * Клас Lab1 представляє собою програму для створення та
 * заповнення квадратної матриці символами.
 * Результати виводяться на екран та записуються в файл
 * "MyFile.txt".
 */
package org.example;

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

```

public class Lab1 {
    public static void main(String[] args) throws
FileNotFoundException {
        int nRows; // Кількість рядків у матриці
        char[][] arr; // Квадратна матриця для зберігання символів
        String filler; // Символ-заповнювач для матриці
        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(); // Зчитуємо розмір матриці від
користувача
        in.nextLine(); // Очищаємо буфер вводу

        // Розраховуємо кількість підмасивів у матриці
        int nSubArrays = (int) Math.ceil((double) nRows / 2.0);

        arr = new char[nSubArrays][]; // Ініціалізуємо масив
підмасивів
        int numElements = nRows;

        // Створюємо підмасиви з відповідною кількістю елементів
        for (int i = 0; i < nSubArrays; i++) {
            arr[i] = new char[numElements];
            numElements -= 2;
            if (numElements < 1) {
                numElements = 1;
            }
        }

        System.out.print("\nВведіть символ-заповнювач: ");
        filler = in.nextLine(); // Зчитуємо символ-заповнювач від
користувача

        int leadingSpaces = 0; // Кількість пробілів перед
СИМВОЛОМ

        exit:
        for (int i = 0; i < nSubArrays; i++) {
            // Додаємо пробіли перед підмасивом
            for (int k = 0; k < leadingSpaces; k++) {
                System.out.print(" ");
                fout.print(" ");
            }

            for (int j = 0; j < arr[i].length; j++) {
                if (filler.length() == 1) {
                    arr[i][j] = (char) filler.codePointAt(0); //
Заповнюємо елементи підмасиву символами-заповнювачами
                    System.out.print(arr[i][j] + " ");
                    fout.print(arr[i][j] + " ");
                } else if (filler.length() == 0) {

```

```

        System.out.print("\nНе введено символ
заповнювач");
        break exit;
    } else {
        System.out.print("\nЗабагато символів
заповнювачів");
        break exit;
    }
}
System.out.print("\n");
fout.print("\n");

    leadingSpaces += 2; // Збільшуємо кількість пробілів
перед наступним підмасивом
}

// Останній елемент матриці
//System.out.println("Останній елемент масиву: " +
arr[nSubArrays - 1][arr[nSubArrays - 1].length - 1]);

    fout.flush(); // Завершуємо запис у файл
    fout.close(); // Закриваємо файл
}
}

```

Результати роботи програми:

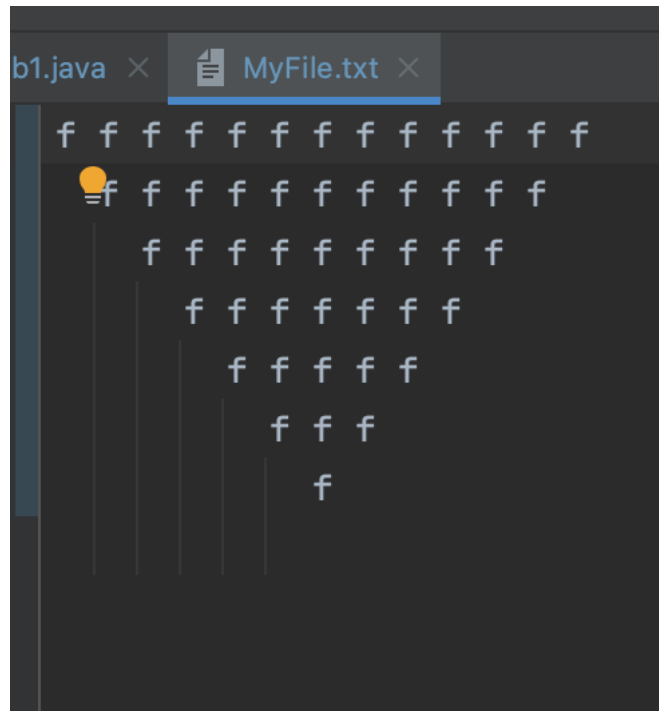
```

Введіть розмір квадратної матриці: 13

Введіть символ-заповнювач: f
f f f f f f f f f f f f f
  f f f f f f f f f f f
    f f f f f f f f f
      f f f f f f f
        f f f f f
          f f f
            f

Process finished with exit code 0

```



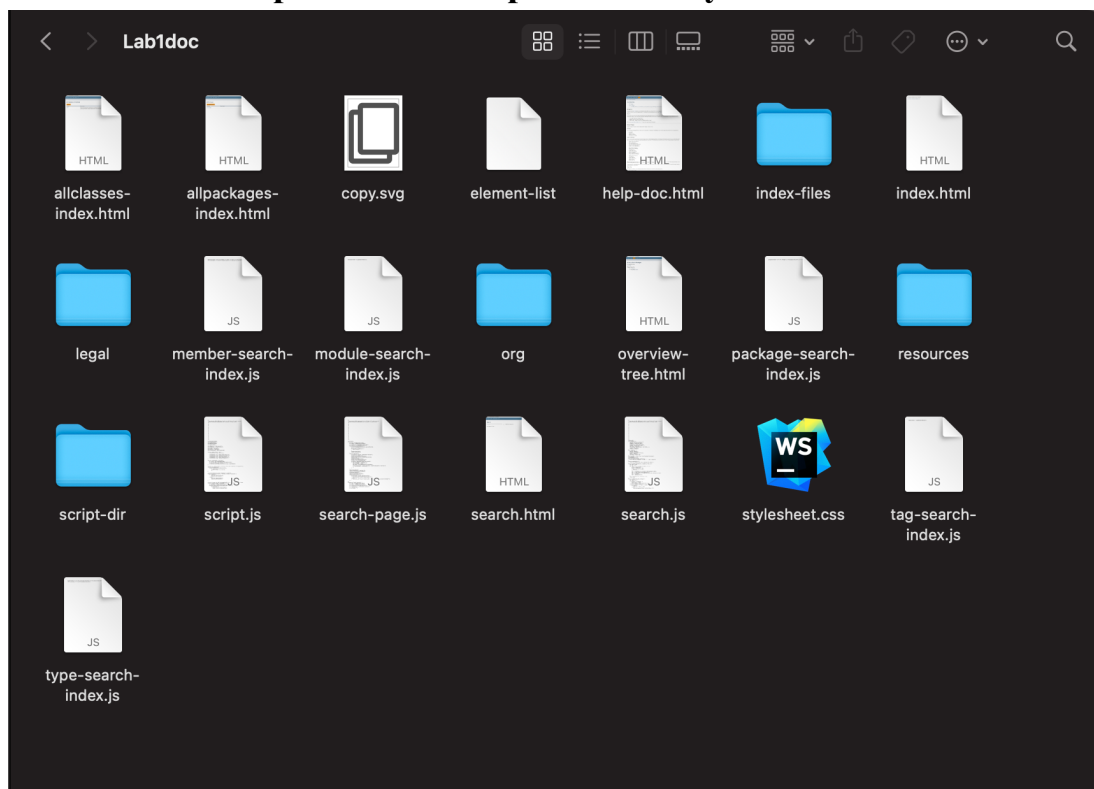
Введіть розмір квадратної матриці: 12

Введіть символ-заповнювач:

Не введено символ заповнювач

Process finished with exit code 0

Фрагмент згенерованої документації



PACKAGE

CLASS

TREE

INDEX

HELP

SUMMARY: NESTED | FIELD | CONSTR | METHOD

DETAIL: FIELD | CONSTR | METHOD

SEARCH

Package org.example

Class Lab1

java.lang.Object

org.example.Lab1

public class Lab1

extends Object

Constructor Summary

Constructors

Constructor	Description
Lab1()	

Method Summary

All Methods

Static Methods

Concrete Methods

Modifier and Type	Method	Description
static void	main(String[] args)	

Methods inherited from class java.lang.Object

equals, getClass, hashCode, notify, notifyAll, toString, wait, wait, wait

Constructor Details

Lab1

public Lab1()

Відповіді на контрольні запитання

- які дескриптори використовуються при коментуванні класів? - @class, @author, @version, та @since
- які дескриптори використовуються при коментуванні методів? - @param, @return, @throws, @deprecated, та інші, що пояснюють параметри, повернене значення, винятки, та інше.
- як автоматично згенерувати документацію? - за допомогою коментарів з дескрипторами та використанням інструментів, таких як JavaDoc.
- які прості типи даних підтримує java? - цілі числа (int, byte, short, long), числа з плаваючою комою (float, double), символи (char), та логічний тип (boolean).
- як оголосити змінну-масив? - тип_даних[] ім'я_масиву; наприклад: int[] numbers;
- які керуючі конструкції підтримує java? - if, else, switch, for, while, та do-while.
- в чому різниця між різними варіантами оператора for? – у синтаксисі та специфіці використання. Наприклад, for зі звичайним лічильником використовується для ітерації з фіксованою кількістю ітерацій, а for-each використовується для ітерації по колекціях.
- як здійснити ввід з консолі? – Для вводу з консолі використовують клас Scanner, наприклад: Scanner scanner = new Scanner(System.in);
- як здійснити ввід з текстового файлу? Для вводу з текстового файлу використовують клас FileReader або Scanner, наприклад: Scanner fin = new Scanner(File("MyFile.txt"));
- як здійснити запис у текстовий файл? Для запису у текстовий файл можна використовувати клас PrintWriter, наприклад: PrintWriter fout = new PrintWriter("MyFile.txt");

Висновок: У ході виконання даної лабораторної роботи, я отримав навички роботи з мовою програмування Java. ознайомився з базовими конструкціями мови, такими як цикли, робота з масивами, ввід та вивід даних з консолі та текстових файлів. Крім того, навчився документувати код, використовуючи коментарі та генерацію документації.