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-306

Ширий Б. І.

Прийняв:

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2022

МЕТОДИЧНІ ВІДОМОСТІ РОБОТИ

МЕТА

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

ЗАВДАННЯ

№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. 1.1);

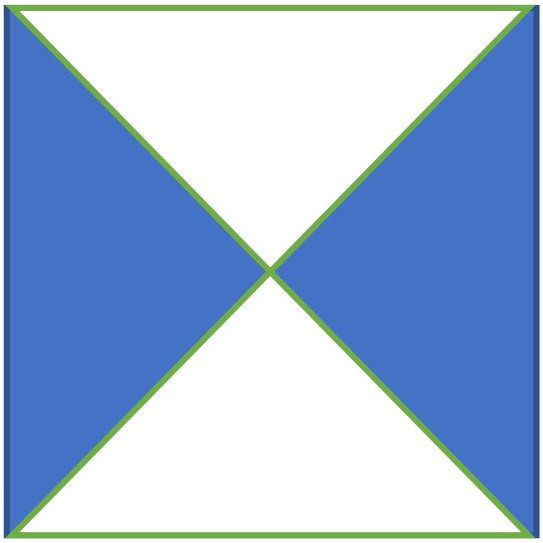

Рисунок 1.1. Заштрихована область квадратної матриці.

- розмір квадратної матриці і символ-заповнювач масиву вводяться з клавіатури;
- при не введені 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?

ВИКОНАННЯ ЛАБОРАТОРНОЇ РОБОТИ

ВИХІДНИЙ КОД

Написав код згідно свого завдання та навів його у лістингу 2.1.

Лістинг 2.1. Код програми.

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

/**
 * The Lab2_Shyryi_CI_306 class implements a program for generating and inferring a jagged array of characters.
 *
 * @author Bohdan Shyryi
 * @version 1.0
 * @since version 1.0
 */
public class Lab2_Shyryi_CI_306 {
	/**
	* The static main method is the entry point into the program
	*
	* @param args
	* @throws FileNotFoundException
	*
	*/
	public static void main(String[] args) throws FileNotFoundException
	{
		String filler;
		Scanner in = new Scanner(System.in);
		File dataFile = new File("Matrix.txt");
		PrintWriter fout = new PrintWriter(dataFile);
		System.out.print("Введіть розмір квадратної матриці: ");
		char[][] arr = new char[in.nextInt()][];
		in.nextLine();
		fout.print("Розмір квадратної матриці: " + arr.length);
		System.out.print("\nВведіть символ-заповнювач: ");
		filler = in.nextLine();
		fout.print("\nСимвол-заповнювач: " + filler);
		if(filler.length() > 1)
		{
			System.out.print("\nВи ввели більше ніж один символ!\nЗавершення програми.");
			fout.print("\nВи ввели більше ніж один символ!\nЗавершення програми.");
			closeProgram(fout);
		}
		if(filler.isEmpty())
		{
			System.out.print("\nВи не ввели символ-заповнювач!\nЗавершення програми.");
			fout.print("\nВи не ввели символ-заповнювач!\nЗавершення програми.");
			closeProgram(fout);
		}

		System.out.print("\n");
		fout.print("\n");

		int num;
		for(int i = 0; i < arr.length; i++)
		{
			if((i + 1)*2 < arr.length)
			{ arr[i] = new char[(i + 1)*2]; }
			else if ((i + 1)*2 > arr.length + 2)
			{ arr[i] = new char[(arr.length - i)*2]; }
			else { arr[i] = new char[arr.length]; }
		}
		System.out.print("\n");

		for(int i = 0; i < arr.length; i++)
		{
			for(int j = 0; j < arr.length; j++)
			{
				num = 0;
				if( (arr[i].length/ 2 > j) || (arr.length / 2 == i) ||
					((arr[i].length/ 2 + j >= arr.length)))
				{
					arr[i][num] = (char)filler.codePointAt(0);;
					System.out.print(arr[i][num] + " ");
					fout.print(arr[i][num] + " ");
					num++;
				}
				else
				{
					System.out.print("  ");
					fout.print("  ");
				}
			}
			System.out.print("\n");
			fout.print("\n");
		}
```

```
                    fout.flush();
                    fout.close();
            }
            /**
            * The closeProgram static method terminates writing to a file and running the program
            *
            * @param args
            *
            */
    private static void closeProgram(PrintWriter fout) {
        fout.flush();
        fout.close();
        System.exit(0);
    }
}
```

РЕЗУЛЬТАТИ ВИКОНАННЯ

Отож, на рисунках 2.1а-б, 2.2-б та 2.3а-б навів результати результати виводу у консоль та у текстовий документ.

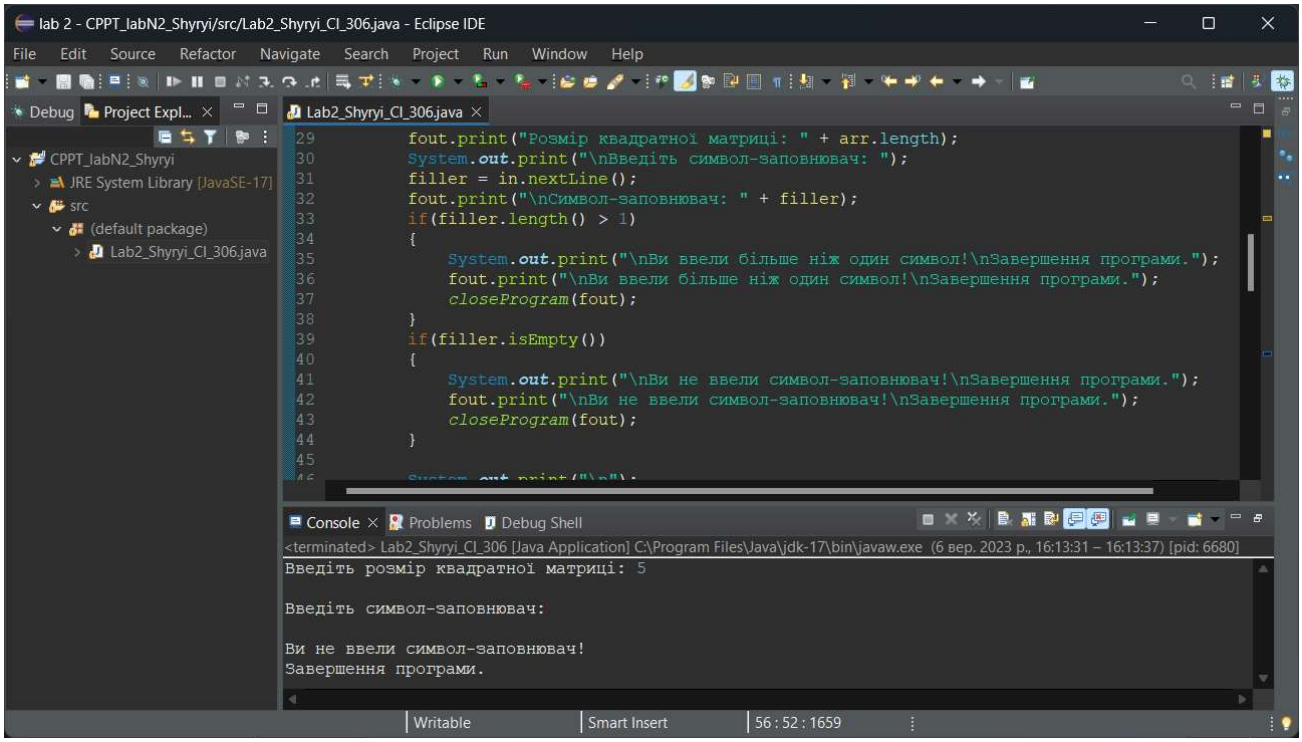

Рисунок 2.1а. Вивід програми у консоль при не введені символа-заповнювача.

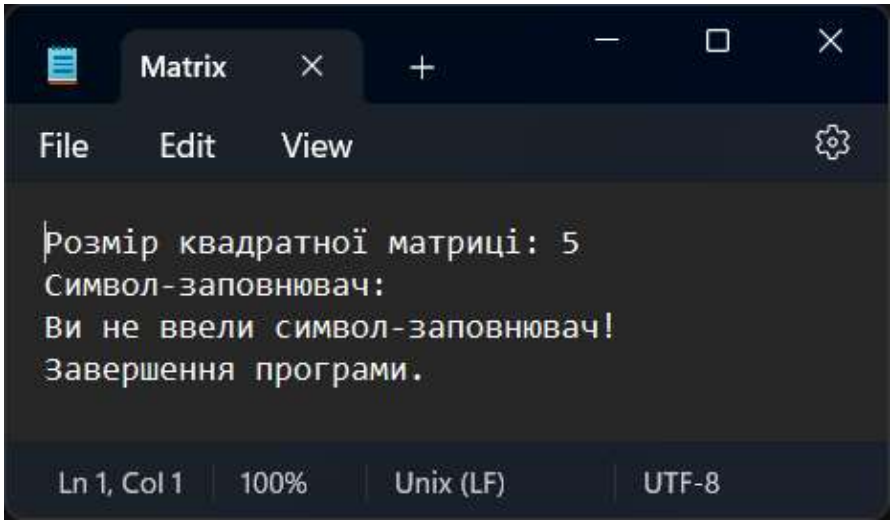

Рисунок 2.1б. Вивід програми у текстовий документ при не введені символа-заповнювача.

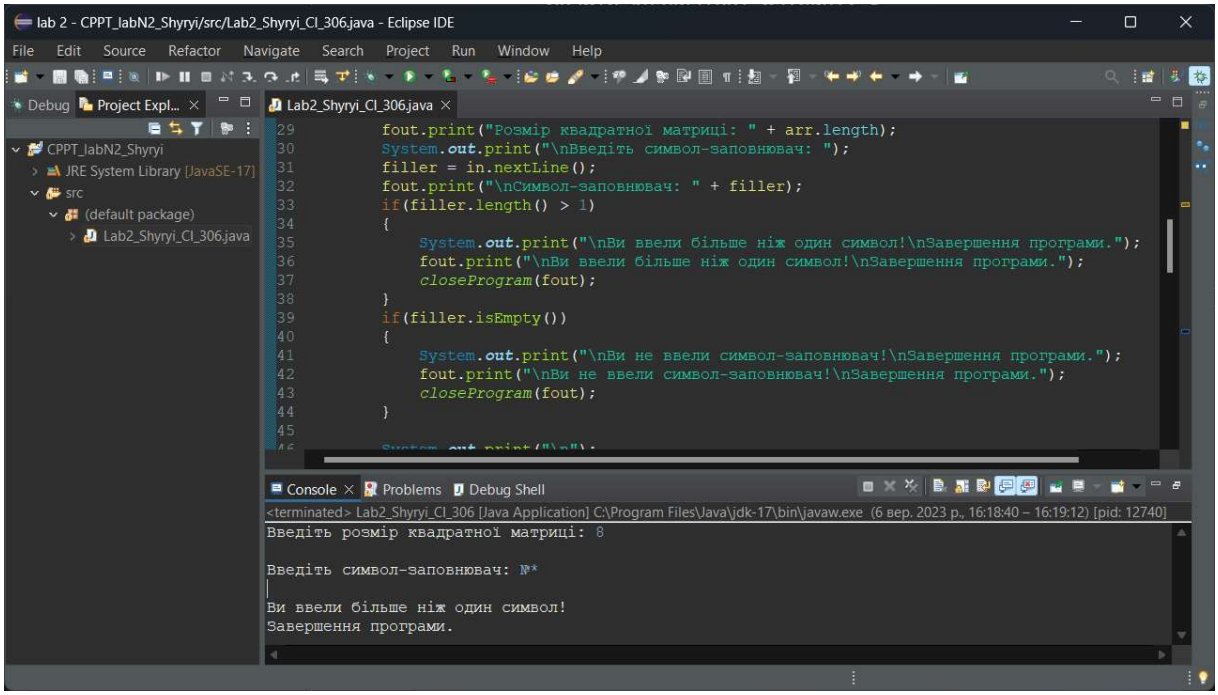

Рисунок 2.2а. Вивід програми у консоль при введені більше одного символа-заповнювача.

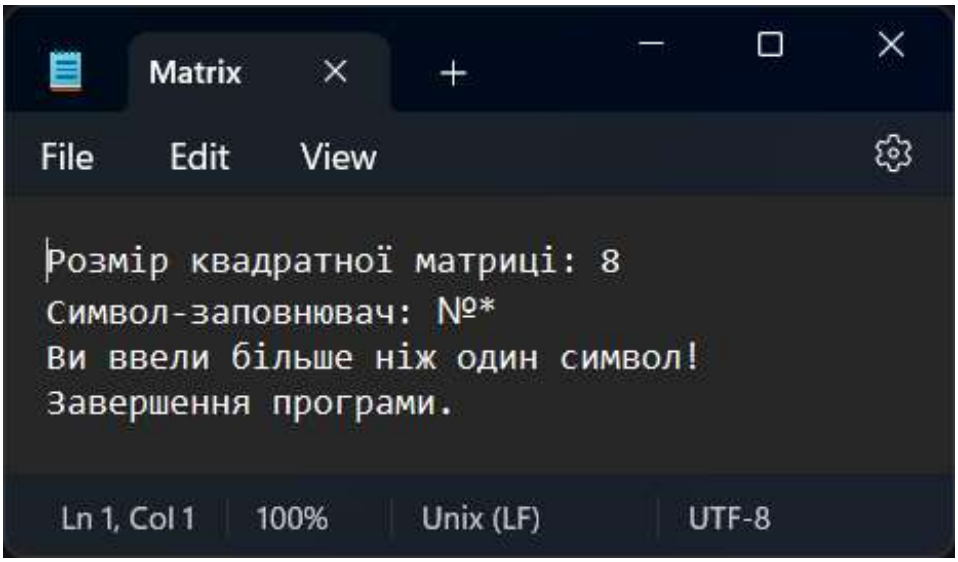

Рисунок 2.2б. Вивід програми у текстовий документ при введені більше одного символа-заповнювача.

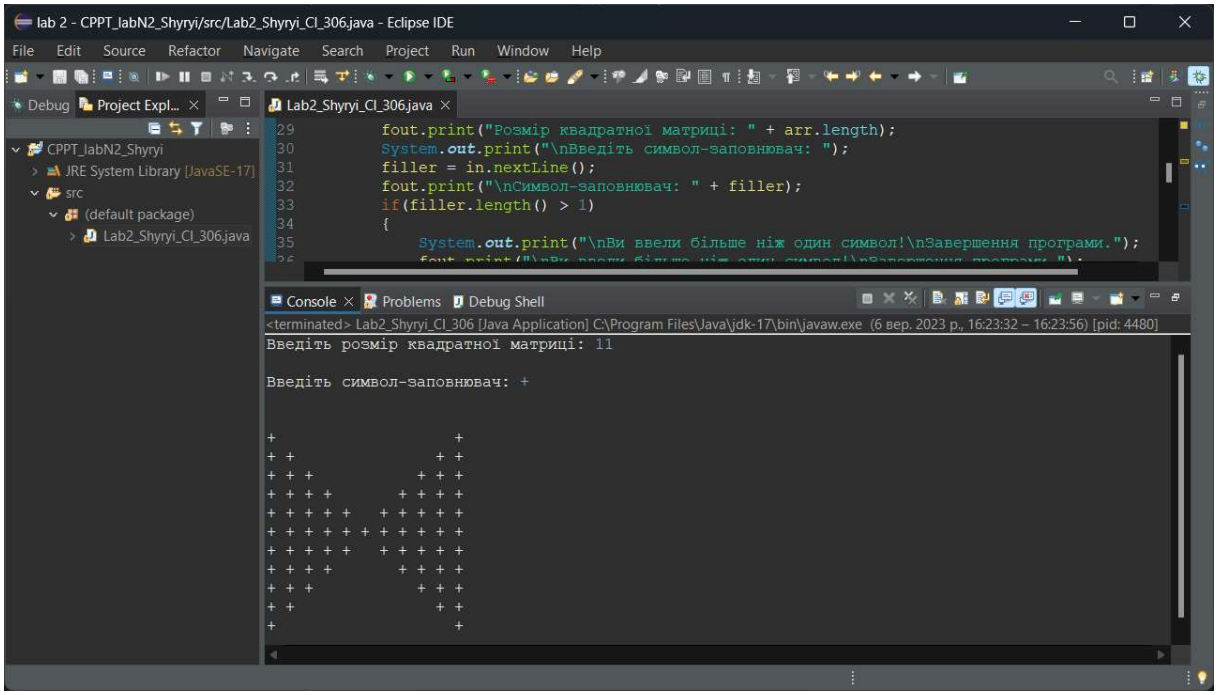

Рисунок 2.3а. Вивід програми у консоль при коректному вводі даних.

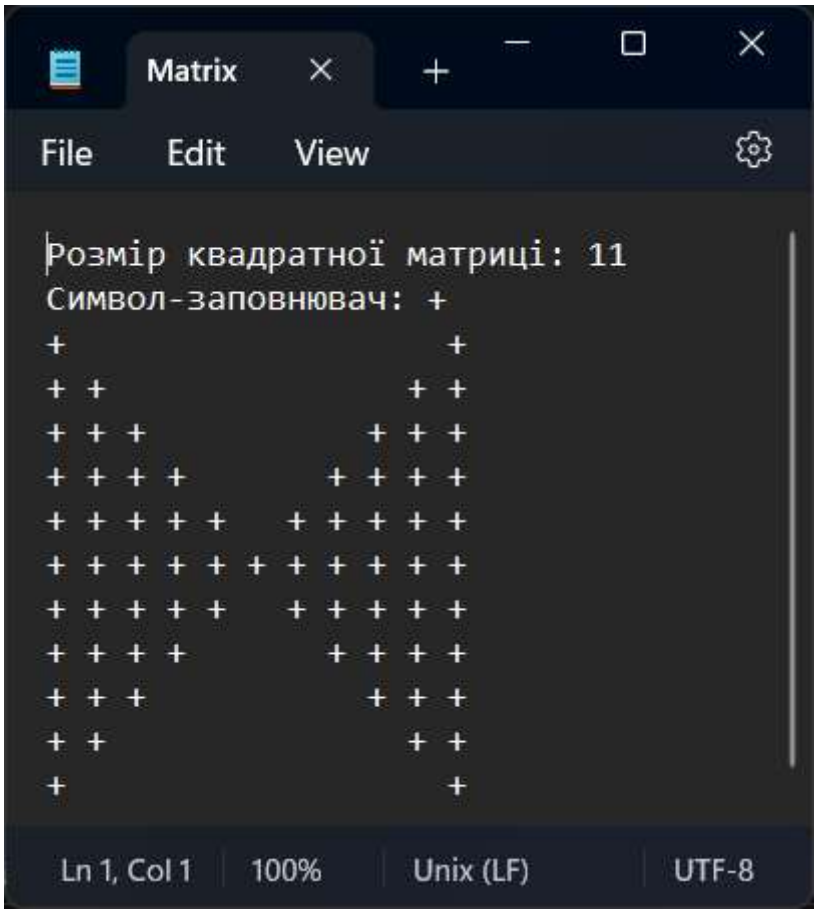

Рисунок 2.3б. Вивід програми у текстовий документ при коректному вводі даних.

ДОКУМЕНТАЦІЯ

На рисунку 2.4 навів фрагмент документації з файлу Lab2_Shyryi_CI_306.html.

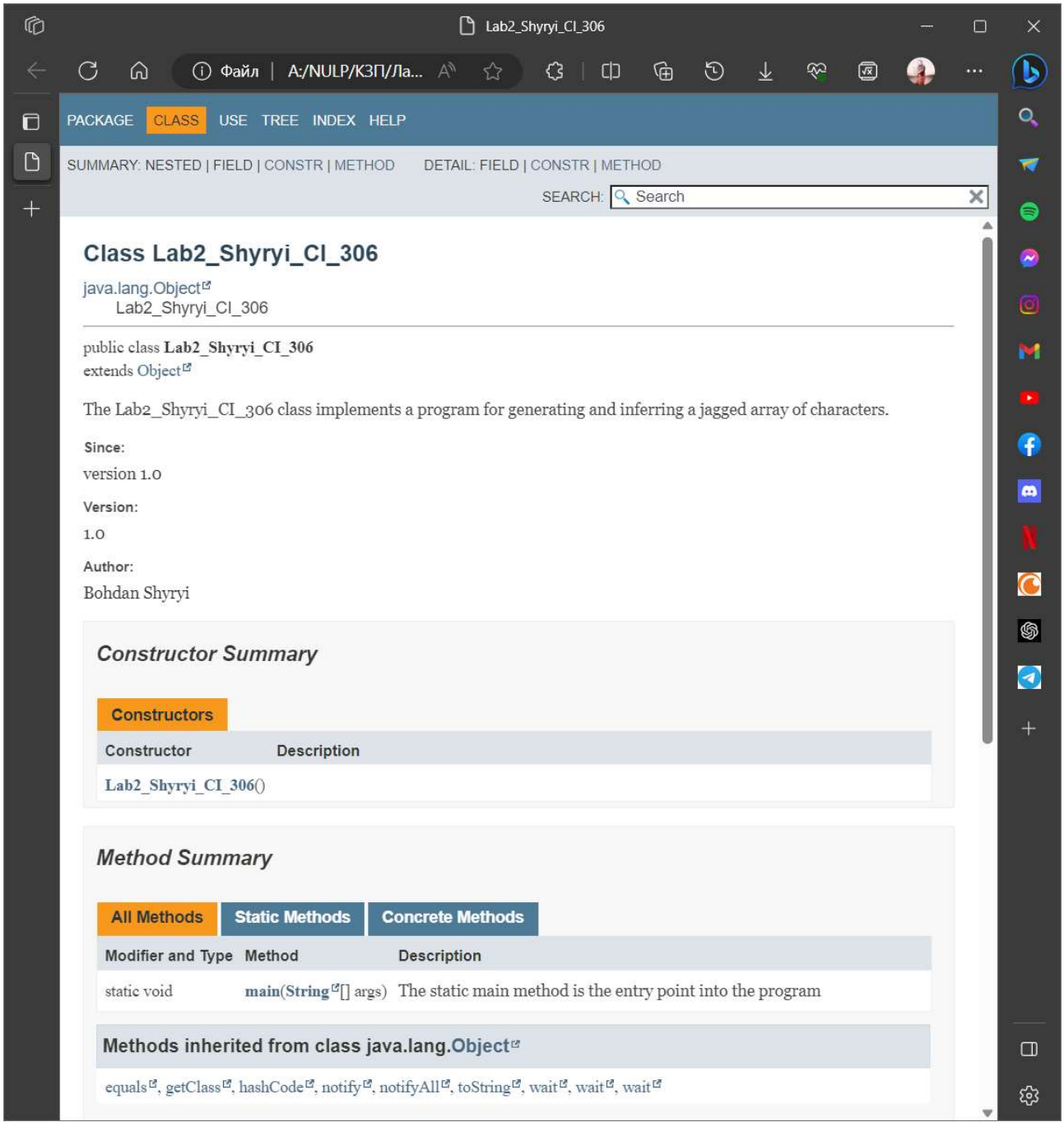

Рисунок 2.4. Фрагмент документації до проекту лабораторної роботи.

ВІДПОВІДІ НА КОНТРОЛЬНІ ПИТАННЯ

ЯКІ ДЕСКРИПТОРИ ВИКОРИСТОВУЮТЬСЯ ПРИ КОМЕНТУВАННІ КЛАСІВ?

Для коментування класів використовуються JavaDoc коментарі, що розміщуються перед оголошенням класу, приклад наведений у лістингу 2.2.

Лістинг 2.2.

```
/**
 * Цей клас представляє об'єкт типу Клас.
 * Він має такі властивості і методи ...
 */
public class Клас {
    // Оголошення класу
}
```

ЯКІ ДЕСКРИПТОРИ ВИКОРИСТОВУЮТЬСЯ ПРИ КОМЕНТУВАННІ МЕТОДІВ?

Також використовуються JavaDoc коментарі, що розміщуються перед оголошенням методу, і містять опис параметрів методу, його поверненого значення і т. д. Наприклад:

Лістинг 2.3.

```
/**
 * Цей метод виконує певну операцію.
 *
 * @param параметр1 Опис параметру 1
 * @param параметр2 Опис параметру 2
 * @return Опис поверненого значення
 */
public int метод(тип параметр1, тип параметр2) {
    // Реалізація методу
}
```

ЯК АВТОМАТИЧНО ЗГЕНЕРУВАТИ ДОКУМЕНТАЦІЮ?

Автоматична генерація документації в Java використовує інструмент JavaDoc, який включений у стандартний комплект поставки Java. Для згенерування документації використовують команду javadoc з командного рядка, і вона зчитує JavaDoc коментарі з вашого коду та створює HTML-документацію.

ЯКІ ПРОСТІ ТИПИ ДАНИХ ПІДТРИМУЄ JAVA?

Прості типи даних, підтримувані в Java, включають **int**, **double**, **float**, **boolean**, **char**, **byte**, **short**, **long**. Також в Java є обгортки (**wrapper classes**) для цих типів, як **Integer**, **Double**, **Boolean**, і т. д.

ЯК ОГОЛОСИТИ ЗМІННУ-МАСИВ?

Змінну-масив можна оголосити так:

тип_даних[] ім'я_масиву;

Наприклад,

int[] numbers;

ЯКІ КЕРУЮЧІ КОНСТРУКЦІЇ ПІДТРИМУЄ JAVA?

Керуючі конструкції, підтримувані в Java, включають **if**, **else**, **switch**, **for**, **while**, і **do-while**.

В ЧОМУ РІЗНИЦЯ МІЖ РІЗНИМИ ВАРІАНТАМИ ОПЕРАТОРА FOR?

Різниця між різними варіантами оператора **for** в Java полягає у синтаксичних особливостях і призначенні для різних сценаріїв. Наприклад, стандартний оператор **for** призначений для ітерації по послідовності, а **enhanced for** (або **for-each**) призначений для ітерації по колекціях і масивах.

ЯК ЗДІЙСНИТИ ВВІД З КОНСОЛІ?

Для введення з консолі в Java використовується клас **Scanner** зі стандартного пакету **java.util**. Спершу потрібно створити об'єкт **Scanner**, а потім використовувати його методи для зчитування даних з консолі.

ЯК ЗДІЙСНИТИ ВВІД З ТЕКСТОВОГО ФАЙЛУ?

Для введення з текстового файлу використовується клас **FileReader** для відкриття файлу і **BufferedReader** для зчитування даних з нього. Також потрібно обробляти винятки, пов'язані з роботою з файлами.

ЯК ЗДІЙСНИТИ ЗАПИС У ТЕКСТОВИЙ ФАЙЛ?

Для запису у текстовий файл використовується клас **FileWriter** для створення або відкриття файлу і **BufferedWriter** для запису даних у файл. Аналогічно, потрібно обробляти винятки, пов'язані з роботою з файлами.

ВИСНОВОК

Протягом лабораторної роботи навчився працювати с масивами у мові програмування Java, а саме із зубчастими. Згідно завдання я зробив створення зубчастого масиву відповідно до мого рисунка 1.1. Спочатку у програмі створюється розмірність масиву, а потім або відбувається присвоювання значень та вивід у консоль та текстовий документ, або виводиться пустота (2 пробіли), оскільки масив не має, що виводити на не відповідній для нього секції матриці. Також, до проекту створив коментарі JavaDoc, які згодом знадобилися для генерації документації документації.