Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Кафедра «Електронних обчислювальних машин»



Звіт

з лабораторної роботи № 5

з дисципліни: «Кросплатформенні засоби програмування»

на тему: « Файли у JAVA»

**Виконав:**

студент групи КІ-306

Фодор Андрій

**Прийняв:**

доцент кафедри ЕОМ

Іванов Ю. С.

Львів – 2023

**Мета роботи:** оволодіти навиками використання засобів мови Java для роботи з потоками і файлами.

**Завдання (варіант № 21)**

1. Створити клас, що реалізує методи читання/запису у текстовому і двійковому форматах результатів роботи класу, що розроблений у лабораторній роботі №4 **(y=sin(3x-5)/ctg(2x))** . Написати програму для тестування коректності роботи розробленого класу.

2. Для розробленої програми згенерувати документацію.

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.

**Вихідний код програми**

package KI306.Fodor.Lab5;

import java.io.\*;

import java.util.\*;

/\*\*

\* Class-driver for demonstration

\*/

public class FileInputOutput {

/\*\*

\* Main method

\* @param args

\*/

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

FileWorkAndCalc obj = new FileWorkAndCalc();

Scanner s = new Scanner(System.in);

System.out.print("Enter x: ");

double data = s.nextDouble();

obj.calculate((int)data);

System.out.println("Result is: " + obj.getResult());

obj.writeResTxt("test1.txt");

obj.writeResBin("test2.bin");

obj.readResBin("test2.bin");

System.out.println("Result from binary file is: " + obj.getResult());

obj.readResTxt("test1.txt");

System.out.println("Result from test file is: " + obj.getResult());

}

}

/\*\*

\* Class FileWorkAndCalc implements calculation, writing to binary and text file and reading from it

\*/

class FileWorkAndCalc {

/\*\*

\* This method allows writing to the file specified in the parameters in txt format

\* @param <code>fName</code> is name of file for writing

\* @throws FileNotFoundException

\*/

public void writeResTxt(String fName) throws FileNotFoundException {

PrintWriter f = new PrintWriter(fName);

f.printf("%f ", result);

f.close();

}

/\*\*

\* Method is used to read from the specified file in txt format

\* @param <code>fName</code> is file name for reading

\*/

public void readResTxt(String fName) {

try {

File f = new File (fName);

if (f.exists()) {

Scanner s = new Scanner(f);

result = s.nextDouble();

s.close();

}

else

throw new FileNotFoundException("File " + fName + "not found");

}

catch (FileNotFoundException ex) {

System.out.print(ex.getMessage());

}

}

/\*\*

\* Method is used to write from the specified file in bin format

\* @param <code>fName</code> is file name for writing

\* @throws FileNotFoundException

\* @throws IOException

\*/

public void writeResBin(String fName) throws FileNotFoundException, IOException {

DataOutputStream f = new DataOutputStream(new FileOutputStream(fName));

f.writeDouble(result);

f.close();

}

/\*\*

\* Method is used to read from the specified file in bin format

\* @param <code>fName</code> is file name for reading

\* @throws FileNotFoundException

\* @throws IOException

\*/

public void readResBin(String fName) throws FileNotFoundException, IOException {

DataInputStream f = new DataInputStream(new FileInputStream(fName));

result = f.readDouble();

f.close();

}

/\*\*

\* This method is used to get result of calculation sin(3x-5)/ctg(2x)

\* @param <code>x</code> is value of degree for calculation

\* @throws CalcException

\*/

public void calculate(int x) throws CalcException {

double y, rad;

rad = x \* Math.PI / 180.0;

try {

y = 1/Math.tan(2\*rad); // ctg(2x)

y = Math.sin(3\*rad-5)/y;

if (y==Double.NaN || y==Double.NEGATIVE\_INFINITY || y==Double.POSITIVE\_INFINITY || x==0 || x== 90)

throw new ArithmeticException();

}

catch (ArithmeticException ex)

{

if (rad==Math.PI/2.0 || rad==0.0)

throw new CalcException("Exception reason: Illegal value of X for cotangent calculation");

else if (Math.tan(2\*rad) == 0 || 1/Math.tan(2\*rad) == 0)

throw new CalcException("Exception reason: Division by 0");

else

throw new CalcException("Unknown reason of the exception during exception calculation");

}

result = y;

}

/\*\*

\* Getter for field result

\* @return result

\*/

public double getResult() {

return result;

}

private double result;

}

/\*\*

\* This class is created to expand opportunities of ArithmeticException class

\*/

class CalcException extends ArithmeticException {

public CalcException(){}

/\*\*

\* This constructor get message, that specifies the exception problem

\* @param <code>cause</code> is text with exception problem

\*/

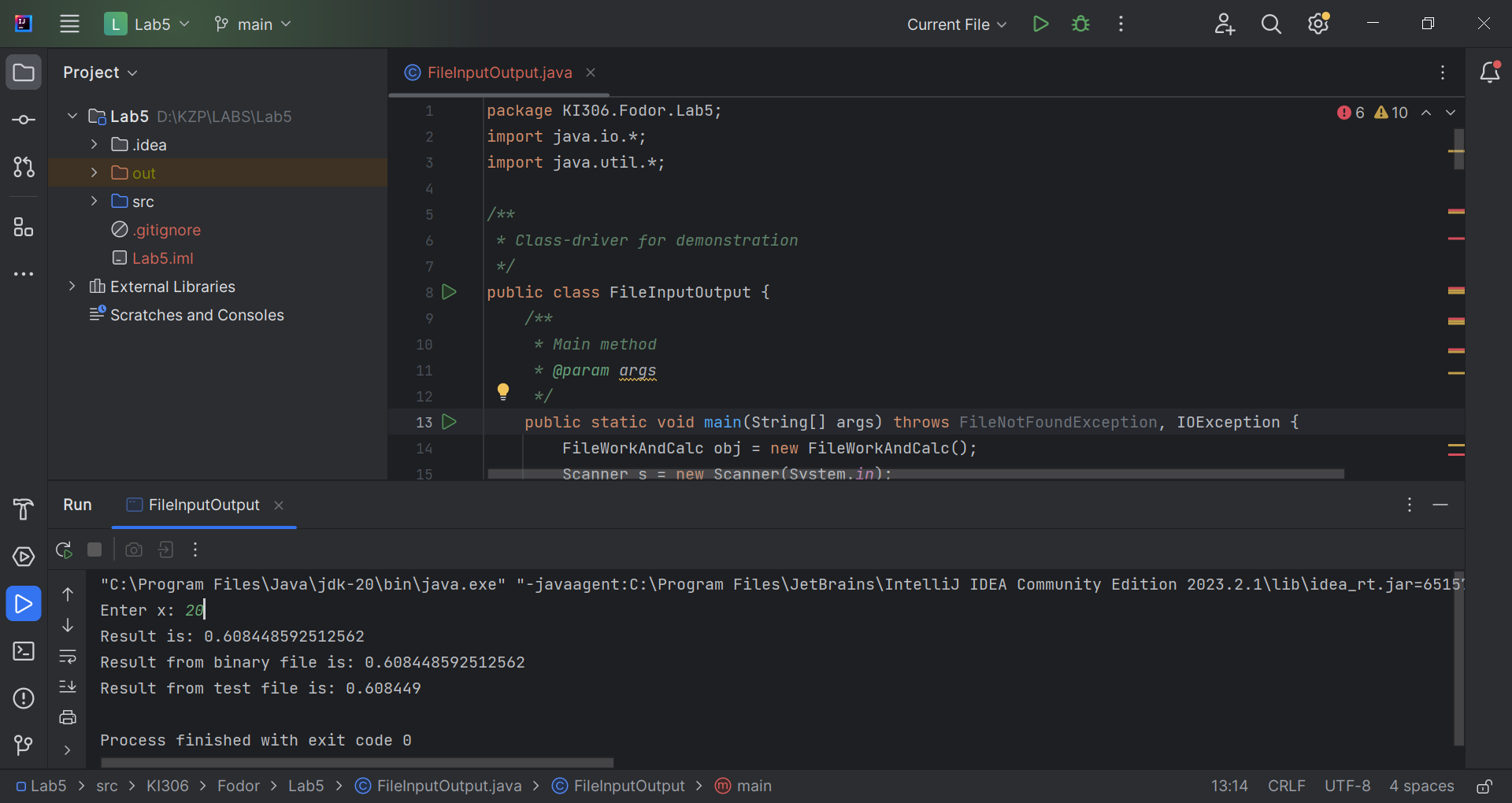
public CalcException(String cause) {

super(cause);

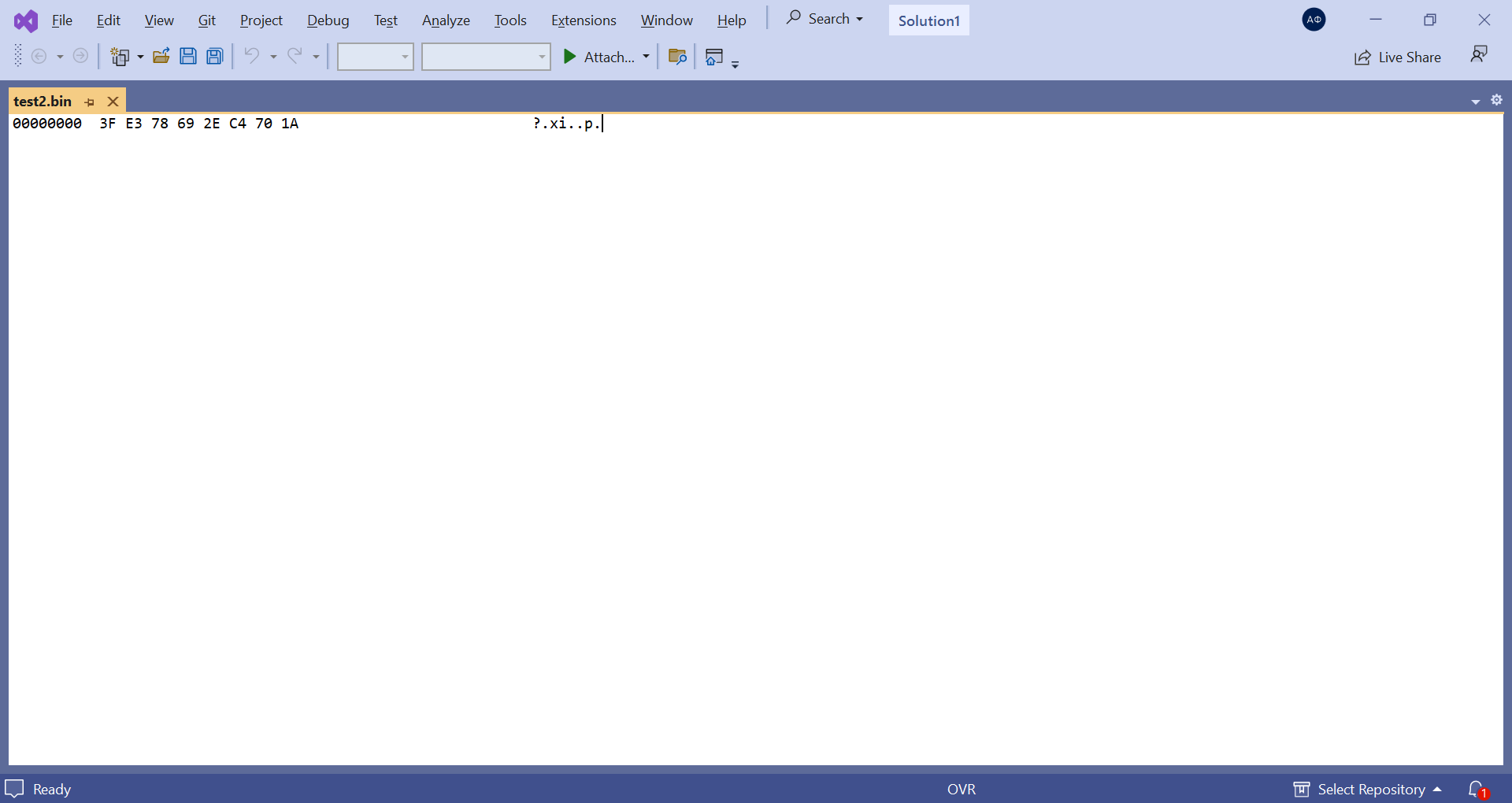
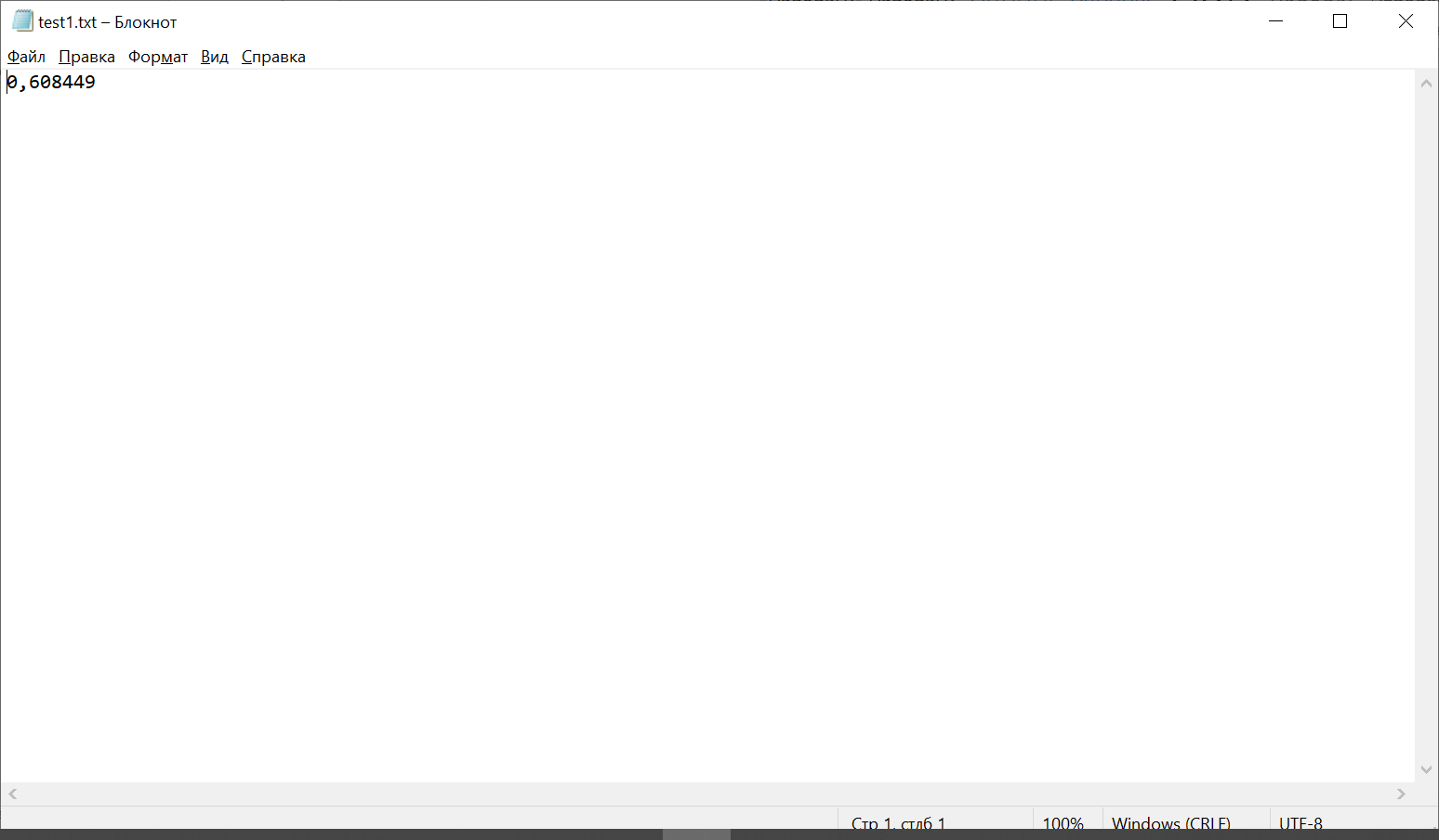
}

}

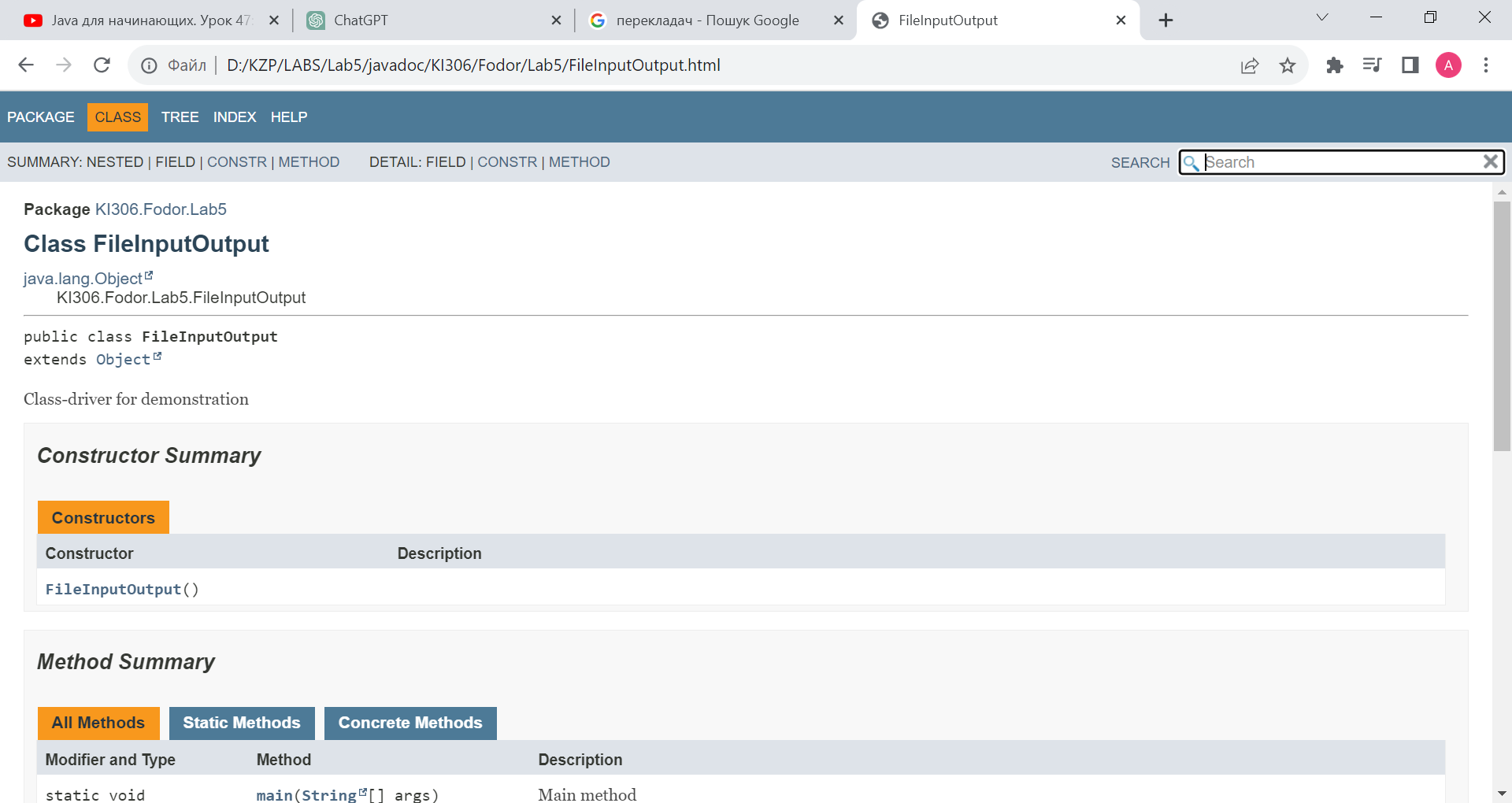
**Результат виконання програми**



Вміст файлу test1.txt та test2.bin після виконання вказаного фрагменту коду:

**Фрагмент згенерованої документації**



**Висновок:** на цій лабораторній роботі я оволодів навиками використання засобів мови Java для роботи з потоками і файлами. Закріпив отримані знання на прикладі виконання індивідуального завдання, створивши клас з методами, що реалізовують безпосереднє обчислення, запис та читання у бінарний та текстовий файл.