

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»

**Виконав:**  
студент групи *КІ-306*  
*Ярема Максим*

**Прийняв:**  
доцент кафедри ЕОМ  
Іванов Ю. С.

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

### Завдання (варіант № 29( $y=\sin(x)/(x+\text{tg}(x)))$ )

1. Створити клас, що реалізує методи читання/запису у текстовому і двійковому форматах результатів роботи класу, що розроблений у лабораторній роботі №5. Написати програму для тестування коректності роботи розробленого класу.
2. Для розробленої програми згенерувати документацію.
3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагмент згенерованої документації.
4. Дати відповідь на контрольні запитання.

### Вихідний код програми

#### Файл EquationsApp.java

```
package ki306.yarema.lab6;

import java.util.Scanner;

/**
 * Class EquationsApp Implements driver for Equations class
 * @author Yarema Maksym
 * @version 1.0
 */
public class EquationsApp {
    /**
     * Method driver
     * @param args
     */
    public static void main(String[] args) {
        Equation equation = new Equation();
        FileUtils fileUtils = new FileUtils();
        double result;
        double fileResult;

        Scanner scanner = new Scanner(System.in);
        System.out.print("Enter X: ");
        try {
            int x = scanner.nextInt();
            result = equation.calculate(x);
            System.out.println("Result is " + result);

            fileUtils.writeResTxt("lab6/textRes.txt", result);
            fileUtils.writeResBin("lab6/binRes.bin", result);

            fileResult = fileUtils.readResBin("lab6/binRes.bin");
            System.out.println("Result from binary file is: " + fileResult);

            fileResult = fileUtils.readResTxt("lab6/textRes.txt");
            System.out.println("Result from txt file is: " + fileResult);
        }
        catch (CalcException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

#### Файл Equation.java

```

package ki306.yarema.lab6;

/**
 * Class Equation implements method for  $y=\sin(x)/(x+\tan(x))$  expression
 * calculation
 * @author Yarema Maksym
 * @version 1.0
 */
public class Equation {
    /**
     * Method calculates the  $y=\sin(x)/(x+\tan(x))$  expression
     * @param x Angle in degrees
     */
    public double calculate(int x) throws CalcException {
        double y, rad;
        rad = x * Math.PI / 180.0;

        try {
            y = Math.sin(x)/(x + Math.tan(x));

            // If the result is not a number, we throw an exception
            if (Double.isNaN(y) || y == Double.NEGATIVE_INFINITY || y ==
Double.POSITIVE_INFINITY || x == 90 || x == -90){
                throw new ArithmeticException();
            }
        }
        catch (ArithmeticException e) {
            // create a higher-level exception with an explanation of the reason for the
error
            if (rad==Math.PI/2.0 || rad==Math.PI/2.0){
                throw new CalcException("Exception reason: Illegal value of X for tangent
calculation");
            }
            else if (x==0) {
                throw new CalcException("Exception reason: X = 0");
            }
            else {
                throw new CalcException("Unknown reason of the exception during exception
calculation");
            }
        }
        return y;
    }
}

```

### Файл CalcException.java

```

package ki306.yarema.lab6;

/**
 * Class CalcException more precises ArithmeticException
 * @author Yarema Maksym
 * @version 1.0
 */
public class CalcException extends ArithmeticException{
    public CalcException(){}
    public CalcException(String cause)
    {
        super(cause);
    }
}

```

### Файл FileUtils.java

```

package ki306.yarema.lab6;

import java.io.*;
import java.util.Locale;
import java.util.Scanner;

/**
 * The FileUtils class provides utility methods for reading and writing double values to
 * files.
 * @author Yarema Maksym
 * @version 1.0
 */
public class FileUtils {
    /**
     * Writes a double value to a text file.
     *
     * @param fName    The name of the file to write to.
     * @param result   The double value to write to the file.
     */
    public void writeResTxt(String fName, double result) {
        try {
            PrintWriter f = new PrintWriter(fName);
            f.print(result);
            f.close();
        }
        catch (FileNotFoundException e) {
            System.out.println("Exception reason: Perhaps wrong file path " + e);
        }
    }

    /**
     * Reads a double value from a text file.
     *
     * @param fName    The name of the file to read from.
     * @return         The double value read from the file, or 0 if the file does not
     * contain a valid double value.
     */
    public double readResTxt(String fName) {
        double result = 0;
        try {
            File file = new File(fName);
            Scanner scanner = new Scanner(file);
            scanner.useLocale(Locale.US);

            if (scanner.hasNextDouble()) {
                result = scanner.nextDouble();
                scanner.close();
            } else {
                System.err.println("The file does not contain a double value.");
            }
        } catch (FileNotFoundException e) {
            System.out.println("Exception reason: Perhaps wrong file path");
        }
        return result;
    }

    /**
     * Writes a double value to a binary file.
     *
     * @param fName    The name of the file to write to.
     * @param result   The double value to write to the file.
     */
    public void writeResBin(String fName, double result) {
        try {
            DataOutputStream f = new DataOutputStream(new FileOutputStream(fName));
            f.writeDouble(result);
            f.close();
        } catch (FileNotFoundException e) {
            System.out.println("Exception reason: Perhaps wrong file path " + e);
        } catch (IOException e) {
            System.out.print(e.getMessage());
        }
    }
}

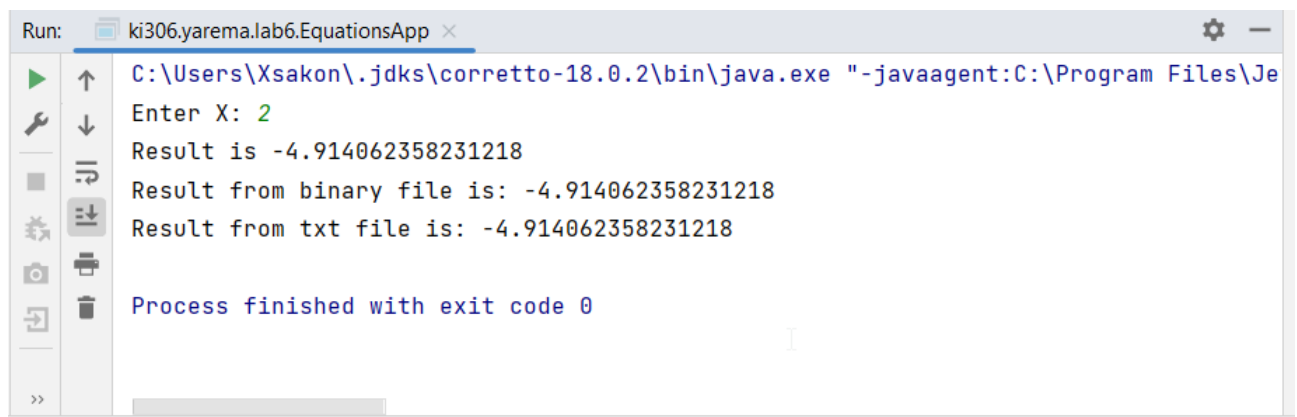
```

```

/**
 * Reads a double value from a binary file.
 *
 * @param fName The name of the file to read from.
 * @return The double value read from the file, or 0 if the file does not
 * contain a valid double value.
 */
public double readResBin(String fName) {
    double result = 0;
    try {
        DataInputStream f = new DataInputStream(new FileInputStream(fName));
        result = f.readDouble();
        f.close();
    } catch (FileNotFoundException e) {
        System.out.println("Exception reason: Perhaps wrong file path " + e);
    } catch (IOException e) {
        System.out.print(e.getMessage());
    }
    return result;
}
}

```

## Результат виконання програми



textRes.txt: -4.914062358231218

binRes.bin: À§ÿö|nN

## Фрагмент згенерованої документації

SEARCH: 

## Package ki306.yarema.lab6

package ki306.yarema.lab6

All Classes and Interfaces	Classes	Exception Classes
Class	Description	
<a href="#">CalcException</a>	Class CalcException more precises ArithmeticException	
<a href="#">Equation</a>	Class Equation implements method for $y=\sin(x)/(x+\operatorname{tg}(x))$ expression calculation	
<a href="#">EquationsApp</a>	Class EquationsApp Implements driver for Equations class	
<a href="#">FileUtils</a>	The FileUtils class provides utility methods for reading and writing double values to files.	



### Відповіді на контрольні запитання

1. Розкрийте принципи роботи з файловою системою засобами мови Java.
  - Для читання і запису файлів використовуються класи, які успадковуються від `InputStream` і `OutputStream` для байтового рівня та `Reader` і `Writer` для текстового рівня.
2. Охарактеризуйте клас `Scanner`.
  - `Scanner` в Java використовується для зчитування вхідних даних, включаючи рядки, числа та інші типи даних з різних джерел, таких як стандартний ввід, файли або рядки
3. Наведіть приклад використання класу `Scanner`.
  - ```
Scanner scanner = new Scanner(System.in);
System.out.print("Enter your name: ");
String name = scanner.nextLine();
System.out.println("Hello, " + name + "!");
```
4. За допомогою якого класу можна здійснити запис у текстовий потік?
  - Для запису в текстовий потік використовується клас `PrintWriter`.

5. Охарактеризуйте клас `PrintWriter`.
  - Клас `PrintWriter` використовується для запису даних у текстовий потік та надає методи для виводу рядків та інших типів даних у текстовому форматі.
6. Розкрийте методи читання/запису двійкових даних засобами мови Java.
  - Для читання та запису двійкових даних використовують класи `DataInputStream` та `DataOutputStream`.
7. Призначення класів `DataInputStream` і `DataOutputStream`.
  - Клас `DataInputStream` використовується для зчитування примітивних типів даних з байтового потоку.
  - Клас `DataOutputStream` використовується для запису примітивних типів даних у байтовий потік.
8. Який клас мови Java використовується для здійснення довільного доступу до файлів.
  - Для довільного доступу до файлів використовується клас `RandomAccessFile`, який дозволяє читати та записувати дані в будь-яку позицію файлу без необхідності читати або записувати дані послідовно.
9. Охарактеризуйте клас `RandomAccessFile`.
  - Для довільного доступу до файлів використовується клас `RandomAccessFile`, який дозволяє читати та записувати дані в будь-яку позицію файлу.
10. Який зв'язок між інтерфейсом `DataOutput` і класом `DataOutputStream`?
  - Клас `DataOutputStream` реалізує інтерфейс `DataOutput`.  
Інтерфейс `DataOutput` визначає методи для запису примітивних типів даних у байтовий потік.  
Клас `DataOutputStream` надає реалізацію цих методів для запису даних у бінарний формат.

## Висновок

У ході виконання даної лабораторної роботи, я отримав навички роботи з засобами мови програмування Java для роботи з потоками і файлами. Ознайомившись з концепцією потоків, я зміг створювати та керувати паралельними виконавчими процесами у моїх програмах. Крім того, я вивчив методи для взаємодії з файловою системою, зчитування та запису даних в текстові файли.