

**АМІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ "ЛЬВІВСЬКА ПОЛІТЕХНІКА"**



ЗВІТ

До лабораторної роботи №6

З дисципліни: «Кросплатформні засоби програмування»

На тему: «ФАЙЛИ»

Варіант 2

Виконав:
ст. гр. КІ-34
Галій О.М.
Перевірив:
доцент
Іванов Ю.С.

Львів 2022

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

ТЕОРЕТИЧНІ ВІДОМОСТІ

Бібліотека класів мови Java має більше 60 класів для роботи з потоками. Потоками у мові Java називаються об'єкти з якими можна здійснювати обмін даними. Цими об'єктами найчастіше є файли, проте ними можуть бути стандартні пристрої вводу/виводу, блоки пам'яті і мережеві підключення тощо. Класи по роботі з потоками об'єднані у кілька ієрархій, що призначені для роботи з різними видами даних, або забезпечувати додаткову корисну функціональність, наприклад, підтримку ZIP архівів.

Класи, що спадкуються від абстрактних класів `InputStream` і `OutputStream` призначені для здійснення байтового обміну інформацією. Підтримка мовою Java одиниць `Unicode`, де кожна одиниця має кілька байт, зумовлює необхідність у іншій ієрархії класів, що спадкується від абстрактних класів `Reader` і `Writer`. Ці класи дозволяють виконувати операції читання/запису не байтних даних, а двобайтних одиниць `Unicode`.

Принцип здійснення читання/запису даних нічим не відрізняється від такого принципу у інших мовах програмування. Все починається з створення потоку на запис або читання після чого викликаються методи, що здійснюють обмін інформацією. Після завершення обміну даними потоки необхідно закрити щоб звільнити ресурси.

Завдання:

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.

$$2. y = \text{ctg}(x)$$

Код програми:

```
package lab6;

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

import static java.lang.System.out;

public class EquationsApp {

    public static void main(String[] args) throws IOException {
        try {
            out.print("Enter file name: ");
            Scanner in = new Scanner(System.in);
            String fName = in.nextLine();
            PrintWriter fout = new PrintWriter(new File(fName));
            try {
                try {
                    CalcWFio obj = new CalcWFio();

                    Equations eq = new Equations();
                    out.print("Enter X: ");
                    int step = in.nextInt();

                    for(int i=1; i<=5; i++) {
                        double result = eq.calculate(step);

                        fout.printf("ctg(%d) = %f\n ",step,result);

                        step = step+25;
                        System.out.println("Result is: " + result);
                        obj.writeResTxt("textRes.txt", result);
                        obj.writeResBin("BinRes.bin", result);

                        obj.readResBin("BinRes.bin");
                        obj.readResTxt("textRes.txt");
                    }

                } finally {
                    // Цей блок виконається за будь-яких обставин
                    fout.flush();
                    fout.close();
                }
            } catch (CalcException ex) {

                // Блок перехоплює помилки обчислень виразу
                out.print(ex.getMessage());
            }
            } catch (FileNotFoundException ex) {

                // Блок перехоплює помилки роботи з файлом навіть якщо вони
```

```

// виникли у блоці finally
    out.print("Exception reason: Perhaps wrong file path");
}
}
}

package lab6;

class Equations {

    public double calculate(int x) throws CalcException {
        double y, rad;
        rad = x * Math.PI / 180.0;
        try{
            y = 1/Math.tan(rad);
            System.out.printf("y = ctg(%d) = %f\n", x , 1/Math.tan(rad));

            // Якщо результат не є числом, то генеруємо виключення
            if (y==Double.NaN || y==Double.NEGATIVE_INFINITY ||
y==Double.POSITIVE_INFINITY || x==90 || x== -90)
                throw new ArithmeticException();
        }
        catch (ArithmeticException ex) {
            // створимо виключення вищого рівня з поясненням причини
            // виникнення помилки
            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;

    }
}

package lab6;

import java.io.DataInputStream;
import java.io.DataOutputStream;
import java.io.File;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Scanner;

class CalcWFio {
//

```

```

public void writeResTxt(String fName, double result) throws FileNotFoundException {
    PrintWriter f = new PrintWriter(fName);
    f.printf("%f", result);
    f.close();
}

```

```

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());
    }
}

```

```

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

    DataOutputStream f = new DataOutputStream(new FileOutputStream(fName));
    f.writeDouble(result);
    f.close();

}

```

```

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

    DataInputStream f = new DataInputStream(new FileInputStream(fName));
    result = f.readDouble();
    f.close();

}

```

```

private double result;
}

```

```

package lab6;

```

```

class CalcException extends ArithmeticException {

```

```

/**
 * Constructor
 */

```

```

public CalcException() {
}

```

```

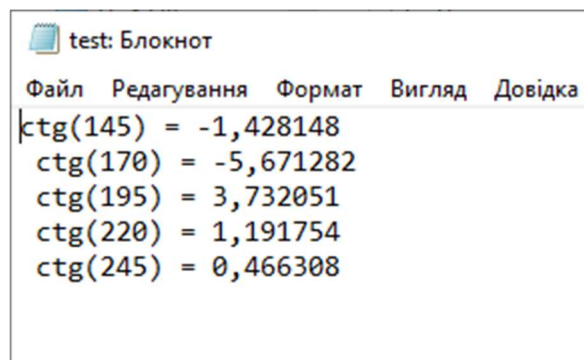
/**
 * Constructor
 *
 * @param cause
 */

```

```
public CalcException(String cause) {  
    super(cause);  
}  
}
```

Результат виконання програми:

```
Enter file name: test  
Enter X: 145  
y = ctg(145) = -1,428148  
Result is: -1.4281480067421137  
y = ctg(170) = -5,671282  
Result is: -5.671281819617712  
y = ctg(195) = 3,732051  
Result is: 3.7320508075688834  
y = ctg(220) = 1,191754  
Result is: 1.19175359259421  
y = ctg(245) = 0,466308  
Result is: 0.46630765815499925
```



test: Блокнот

Файл Редагування Формат Вигляд Довідка

ctg(145) = -1,428148
ctg(170) = -5,671282
ctg(195) = 3,732051
ctg(220) = 1,191754
ctg(245) = 0,466308

Висновок: під час лабораторної роботи я оволодів навичками використання засобів мови Java для роботи з потоками і файлами.