



Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

Лабораторна робота
№3
Управління потоком виконання

Виконали
Студенти групи ІА-23:
Ковальський К.В., Щипун М.Г.

Перевірив:
Колеснік В.М.

Київ 2022

Розподіл варіантів:

Хід роботи:

Завдання 3(Ковальський):

3

Обчислити суму ряду: $\sum_{i=1}^k \left(\frac{1}{\sqrt{z \cdot i}} + \operatorname{tg}\left(\frac{k}{i}\right) \right), \quad k < 15$

```
task1.java x task2.java x task3.java x
1 public class task1 {
2     public static void main(String[] args){
3         printResults( k: 13, z: 4, i: 1);
4         printResults( k: 15, z: 3, i: 7);
5         printResults( k: 16, z: 8, i: 1);
6         printResults( k: 3, z: 2, i: 1);
7         printResults( k: 9, z: 13, i: 10);
8         printResults( k: 11, z: 12, i: 13);
9         printResults( k: 14, z: 3, i: 13);
10    }
11    1 usage
12    public static double function(int k, double z, double i){
13        double result;
14        result = (1/Math.sqrt(z*i) + Math.tan(k/i));
15        return result;
16    }
17    1 usage
18    public static double sum(int k, double z, double i){
19        double current;
20        int count = 1;
21        double sum=0;
22        sum += count;
23        count++;
24    }
```

```

25     if(k>15){
26         throw new IllegalArgumentException("param k>15");
27     }
28     while(count<k){
29         current = function(k,z,i);
30         sum += current;
31         count++;
32     }
33     return sum;
34 }
35
36 7 usages
37 static void printResults(int k, double z, double i) {
38     System.out.print("k:" + k + "z:" + z + " " + "i:" + i + "sum: ");
39     try {
40         System.out.println(sum(k,z,i));
41     } catch (IllegalArgumentException e){
42         System.out.println("EXCEPTION! " + e.getMessage());
43     }
44 }

```

```

C:\Users\kovalskii\.jdk\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program Files\
k:13z:4.0 i:1.0sum: 11.593232462301387
k:15z:3.0 i:7.0sum: -16.353268578944817
k:16z:8.0 i:1.0sum: EXCEPTION! param k>15
k:3z:2.0 i:1.0sum: 1.5645602381122696
k:9z:13.0 i:10.0sum: 10.435048136367294
k:11z:12.0 i:13.0sum: 11.886447094721673
k:14z:3.0 i:13.0sum: 25.21088931454905

```

Завдання 9(Щипун):

9	$x(t,l) = \sum_{i=1}^t \begin{cases} \sqrt{t \cdot l}, & l = 1, 3, 5, \dots \\ \frac{l}{\sqrt{t}}, & l = 2, 4, 6, \dots \end{cases}$ <p>Обчислити значення функції:</p>
---	---

```

task1.java × task2.java × task3.java ×
1  import static java.lang.Math.*;
2  public class task2 {
3      public static void main(String[] args){
4          printResults(t: 1, l: 1);
5          printResults(t: 2, l: 1);
6          printResults(t: 1, l: -1);
7          printResults(t: -1, l: 2);
8          printResults(t: 25, l: 20);
9          printResults(t: 1300, l: 300);
10     }
11     1 usage
12     public static double function(int t, int l){
13         double sum = 0;
14         for(int i=0; i<t; i++){
15             if(l % 2 == 1){
16                 sum = sqrt(t*l);
17             }else{
18                 sum = l / sqrt(t);
19             }
20         }return sum;
21     6 usages
22     static void printResults(int t, int l){
23         System.out.println("t:" + t + "l:" + l + "result:" + function(t, l));
24     }
25 }

```

```

C:\Users\kovalskii\.jdk\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program
t:1l:1result:1.0
t:2l:1result:1.4142135623730951
t:1l:-1result:-1.0
t:-1l:2result:0.0
t:25l:20result:4.0
t:1300l:300result:8.320502943378438

```

Завдання 15(Ковальський):

15

Обчислити нескінченну суму
$$\sum_{i=1}^{\infty} \frac{(-1)^{i+1}}{i \cdot (i+1) \cdot (i+2)}$$
 із заданою точністю ϵ ($\epsilon > 0$).
Вважати, що необхідна точність досягнута якщо черговий доданок виявився по модулю меншим, ніж ϵ . Цей і усі наступні доданки можна не враховувати.

```
task1.java × task2.java × task3.java ×
1 import static java.lang.Math.*;
2 public class task3 {
3
4     public static void main(String[] args) {
5         printResults(e: -1);
6         printResults(e: 0);
7         printResults(e: 1);
8         printResults(e: 2);
9         printResults(e: 20);
10        printResults(e: 0.01);
11        printResults(e: 0.0001);
12        printResults(e: 0.000001);
13        printResults(e: 0.000000001);
14        printResults(Double.NaN);
15        printResults(Double.MIN_VALUE);
16        printResults(Double.MAX_VALUE);
17    }
18
19    1 usage
20    public static double task15(double e) {
21        if ((e <= 0) || (Double.isNaN(e))) {
22            throw new IllegalArgumentException("param e (accuracy) = " + e);
23        }
24        double sumResult = 0.0;
25        double iterationVal = e;
26        double powVal = -1;
27        for(int i = 1; i < Integer.MAX_VALUE; ++i)
28        {
29            powVal *= -1;
30            iterationVal = powVal/(i*(i+1)*(i+2));
```

```
            sumResult += iterationVal;
            if (Math.abs(iterationVal) < e) break;
        }
        if (Math.abs(iterationVal) >= e) {
            throw new RuntimeException("Result accuracy not less than param e, actual accuracy is: " + iterationVal);
        }
        return sumResult;
    }
12 usages
    static void printResults(double e) {
        System.out.print("e: " + e + " result: ");
        try {
            System.out.println(task15(e));
        } catch (Exception ex) {
            System.out.println("EXCEPTION! " + ex.getMessage());
        }
    }
}
```

```
C:\Users\kovalskii\.jdk\openjdk-19.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBrains\IntelliJ IDEA
e: -1.0 result: EXCEPTION! param e (accuracy) = -1.0
e: 0.0 result: EXCEPTION! param e (accuracy) = 0.0
e: 1.0 result: 0.16666666666666666
e: 2.0 result: 0.16666666666666666
e: 20.0 result: 0.16666666666666666
e: 0.01 result: 0.13333333333333333
e: 1.0E-4 result: 0.13633821384191558
e: 1.0E-6 result: 0.13629388298459158
e: 1.0E-9 result: 0.13629436062213426
e: NaN result: EXCEPTION! param e (accuracy) = NaN
e: 4.9E-324 result: EXCEPTION! Result accuracy not less then param e, actual accuracy is: -Infinity
e: 1.7976931348623157E308 result: 0.16666666666666666

Process finished with exit code 0
```