



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

Лабораторна робота №3
із дисципліни *«Основи програмування»*
Тема: «Управління потоком виконання»

Виконали:
Студенти групи ІА-24
Чайка Антон Павлович
Коханчук Михайло Миколайович
Котлярчук Максим Сергійович

Перевірив:
Колеснік Валерій Миколайович

1. Обчислити суму ряду: $\sum_{i=1}^k \sqrt{m \cdot \frac{1}{i}} \cdot \sin(m \cdot i), \quad k \leq 30$

```
public class Task1 {
    public Task1() {
    }

    public static void main(String[] args) {
        printResults(m: 0.0, k: 0);
        printResults(m: 0.0, k: 1);
        printResults(m: 1.0, k: 0);
        printResults(m: 1.0, k: 1);
        printResults(m: 2.0, k: 1);
        printResults(m: 1.0, k: 2);
        printResults(m: 2.0, k: 40);
        printResults(m: 30.0, k: 30);
        printResults(m: 31.0, k: 31);
        printResults(m: 1.0, k: 10);
        printResults(m: 32.0, k: 10);
        printResults(m: 0.0, Integer.MAX_VALUE);
        printResults(m: 1.0, Integer.MAX_VALUE);
    }

    public static double axx(double m, int k) {
        double expression = 0.0;
        if (k > 30) {
            throw new IllegalArgumentException("param k = " + k);
        } else {

```

```
        double expression = 0.0;
        if (k > 30) {
            throw new IllegalArgumentException("param k = " + k);
        } else {
            for(int i = 0; i <= k; ++i) {
                expression = Math.sqrt(m / (double)i) * Math.sin(m * (double)i);
            }

            return expression;
        }
    }

    static void printResults(double m, int k) {
        System.out.print("m:" + m + " x:" + k + " result:");

        try {
            System.out.println(axx(m, k));
        } catch (IllegalArgumentException var4) {
            System.out.println("EXCEPTION! " + var4.getMessage());
        }
    }
}
```

Результат:

```
m:0.0 x:0 result:NaN
m:0.0 x:1 result:0.0
m:1.0 x:0 result:NaN
m:1.0 x:1 result:0.8414709848078965
m:2.0 x:1 result:1.2859407532478362
m:1.0 x:2 result:0.6429703766239181
m:2.0 x:40 result:EXCEPTION! param k = 40
m:30.0 x:30 result:0.9978032744219705
m:31.0 x:31 result:EXCEPTION! param k = 31
m:1.0 x:10 result:-0.17203458056254387
m:32.0 x:10 result:-0.765907713705885
m:0.0 x:2147483647 result:EXCEPTION! param k = 2147483647
m:1.0 x:2147483647 result:EXCEPTION! param k = 2147483647

Process finished with exit code 0
```

$$x(t, i) = \begin{cases} \sqrt{t}, & i = 1 \\ \frac{1}{\sqrt{t}}, & i = 2 \\ \sum_{k=1}^i k \cdot t, & i \neq 1, i \neq 2 \end{cases}$$

6. Обчислити значення функції:

```

public class Task6 {
    public Task6() {}

    public static void main(String[] args) {
        printResults(1.0, 1);
        printResults(2.0, 2);
        printResults(4.0, 3);
        printResults(4.0, 4);
        printResults(0.0, 1);
        printResults(0.0, 2);
        printResults(0.0, 3);
        printResults(-1.0, 1);
        printResults(-1.0, 2);
        printResults(-1.0, 3);
        printResults(Double.NaN, 1);
        printResults(4.0, Integer.MAX_VALUE);
    }

    public static double axx(double t, int i) {
        if (i < 1) {
            throw new IllegalArgumentException("param i = " + i);
        } else {
            double expression = 0.0;
            switch (i) {

```

```

                expression = Math.sqrt(t);
            case 2:
                expression = 1.0 / Math.sqrt(t);
            default:
                for(int k = 0; k <= i; ++k) {
                    expression += (double)k * t;
                }
            return expression;
        }
    }

    static void printResults(double t, int i) {
        System.out.print("i: " + i + " t: " + t + " result: ");

        try {
            System.out.println(axx(t, i));
        } catch (IllegalArgumentException var4) {
            System.out.println("EXCEPTION! " + var4.getMessage());
        }
    }
}

```

Результат:

```

i:1 t: 1.0 result: 2.0
i:2 t: 2.0 result: 6.707106781186548
i:3 t: 4.0 result: 24.0
i:4 t: 4.0 result: 40.0
i:1 t: 0.0 result: Infinity
i:2 t: 0.0 result: Infinity
i:3 t: 0.0 result: 0.0
i:1 t: -1.0 result: NaN
i:2 t: -1.0 result: NaN
i:3 t: -1.0 result: -6.0
i:1 t: NaN result: NaN
i:2147483647 t: 4.0 result:

```

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

```

public static void main(String[] args) {
    printResults(eps: 0.0);
    printResults(eps: 1.0);
    printResults(eps: 10.0);
    printResults(eps: 11.0);
    printResults(eps: 21.0);
    printResults(eps: 12.0);
    printResults(eps: 40.0);
    printResults(eps: 3.0);
    printResults(eps: 31.0);
    printResults(eps: 2.0);
    printResults(eps: 4.0);
    printResults(eps: 2.147483647E9);
    printResults(eps: 0.4);
}

1 usage
public static double axx(double eps) {
    double sum = 0.0;
    int i = 0;
    ++i;
    sum += (double)(1 / (i * i));
    if (eps <= Math.abs(1.0 / Math.pow((double)i, 2.0))) {
        throw new IllegalArgumentException("param eps = " + eps);
    } else {

```

```

    } else {
        return sum;
    }
}

13 usages
static void printResults(double eps) {
    System.out.print("Epsilon: " + eps + " Summary: ");

    try {
        System.out.println(axx(eps));
    } catch (IllegalArgumentException var3) {
        System.out.println("EXCEPTION! " + var3.getMessage());
    }
}
}

```

Результат:

```
Epsilon: 0.0 Summary: EXCEPTION! param eps = 0.0  
Epsilon: 1.0 Summary: EXCEPTION! param eps = 1.0  
Epsilon: 10.0 Summary: 1.0  
Epsilon: 11.0 Summary: 1.0  
Epsilon: 21.0 Summary: 1.0  
Epsilon: 12.0 Summary: 1.0  
Epsilon: 40.0 Summary: 1.0  
Epsilon: 3.0 Summary: 1.0  
Epsilon: 31.0 Summary: 1.0  
Epsilon: 2.0 Summary: 1.0  
Epsilon: 4.0 Summary: 1.0  
Epsilon: 2.147483647E9 Summary: 1.0  
Epsilon: 0.4 Summary: EXCEPTION! param eps = 0.4
```

Висновки:

Під час виконання даної лабораторної роботи, ми навчилися використовувати цикли в програмі, обробляти виключення та обчислювати скінченну і нескінченну суми.