

МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ  
УЧЕРЕЖДЕНИЕ ОБРАЗОВАНИЯ  
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»  
КАФЕДРА ИИТ

Лабораторная работа №1  
по дисциплине «Современные платформы программирования»

Выполнила:  
Андросюк М.М.  
Группа: ПО-5  
Проверил:  
Крощенко А.А.

Брест 2022

**Цель работы:** приобрести навыки написания простого многопоточного приложения с использованием Java API.

**Задание:** разработать приложение с использованием Java API, использующее один вспомогательный поток, вычисляющий заданную сумму и выполняющий вывод результата вычисления (как конечный, так и промежуточные). В программе должны быть предусмотрены функции приостановки, возобновления и полной остановки выполнения потока с выводом соответствующего сообщения. В случае быстрого выполнения потока и, как следствие, невозможности демонстрации функций приостановки, продумать искусственное «торможение» потока для достижения заданных целей. Обработать исключения.

**Текст программы:**

```
package com.company;
import
java.util.Scanner;
public class Main {
    private static Thread thread;
    private static Double result;
    private final static Object lock = new
    Object(); private static boolean paused;

    public static void main(String[] args) {
        Scanner in = new Scanner(System.in);

        while (true) {
            String valueStr = in.nextLine();

            if
                (valueStr.equals("Pause"
                )) { pause();
                continue;
            }

            if (valueStr.equals("Resume"))
                { resume();
                continue;
            }

            if
                (valueStr.equals("Stop"
                )) { stop();
                continue;
            }
            int
            value;
            try {
                value = Integer.parseInt(valueStr);
```

```

        if (value < 0) {
            throw new IllegalArgumentException();
        }
    }
    catch (IllegalArgumentException
        e) {
        System.out.println("Invalid
            input"); return;
    }

    calculate(value);
}

private static void calculate(int
    value) { if (thread != null) {
        thread.interrupt();
    }

    result = 0.0;
    paused = false;
    System.out.println("Started"
    ); thread = new Thread(() -
    > {
        double previousTerm = 1.0;

        for (int i = 0; i <= value; i++) {
            synchronized (lock) {
                try {
                    Thread.sleep(10000);

                    if (paused) {
                        lock.wait();
                    }

                    previousTerm = i == 0 ? 1.0 : previousTerm /
                    i; result += previousTerm;

                    System.out.println(result);
                } catch (InterruptedException e)
                { break;
                }
            }
        }
    }
    });

    thread.start();
}

protected static void pause() {
    if (thread != null &&
        thread.isAlive()) { paused = true;

```

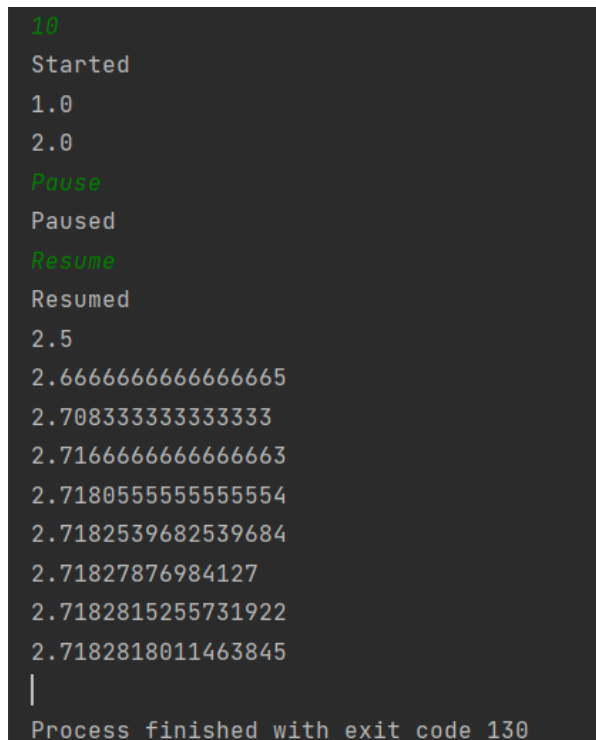
```

        System.out.println("Paused");
    }
}

protected static void resume() {
    if (paused && thread != null && thread.isAlive()) {
        synchronized (lock) {
            paused = false;
            lock.notify();
            System.out.println("Resumed");
        }
    }
}

protected static void
stop() { if (thread !=
null) {
    thread.interrupt();
    System.out.println("Stopped");
}
}
}

```



```

10
Started
1.0
2.0
Pause
Paused
Resume
Resumed
2.5
2.6666666666666665
2.708333333333333
2.7166666666666663
2.7180555555555554
2.7182539682539684
2.71827876984127
2.7182815255731922
2.7182818011463845
|
Process finished with exit code 130

```

**Вывод:** приобретены навыки написания простого многопоточного приложения с использованием Java API.