

Министерство образования Республики Беларусь
Учреждение образования
«Брестский государственный технический университет»
Кафедра интеллектуальных информационных технологий

Тема: «Многопоточное приложение»

Отчёт лабораторной работы №1
по дисциплине «Современные платформы программирования»
за II семестр

Выполнил:
студент 3-его курса
VI-го семестр
факультета ЭИС
группы ПО-4(1)
зачётная книжка №190333
Галанин П. И.
«__» _____ 2022 г.

Проверил:
ассистент
кафедры ИИТ
Монтик Н. С.
«__» _____ 2022 г.

Отчёт лабораторной работы №1

Тема: «Многопоточное приложение»

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

Что нужно сделать:

Разработать консольное приложение с двумя потоками (1 - основной поток; 2 - поток, в котором производится вычисление).

Вариант 5:
$$\sum_{k=1}^n \frac{1}{(2k-1)(2k+1)} = \frac{1}{1*3} + \frac{1}{3*5} + \dots + \frac{1}{(2n-1)(2n+1)}$$

Исходный код:

Листинг: src/com/company/Main.java

```
1 package com.company;
2
3 public class Main {
4     public static void main(String[] args) {
5         System.out.println("Main thread created");
6         MyThread class_instance = new MyThread();
7         class_instance.start();
8
9         try {
10             class_instance.join();
11         }
12         catch (Exception e) {
13             System.out.println(e);
14         }
15
16         System.out.println("Main thread finished");
17     }
18 }
```

Листинг: src/com/company/MyThread.java

```
1 package com.company;
2
3 import java.util.Scanner;
4
5 public class MyThread extends Thread {
6     public void run() {
7         System.out.println("Child thread created");
8         Scanner in = new Scanner(System.in);
9
10        System.out.print("n = ");
11        int n = in.nextInt();
12
13        double a;
14        double sum = 0;
15        for (int i = 1; i < n; ++i) {
16            a = (double) 1 / ( (2*i - 1) * (2 * i + 1) );
17            //System.out.println(a);
18            sum += a;
19        }
20    }
```

```
21         System.out.println("sum = " + sum);
22         System.out.println("Child thread finished");
23     }
24 }
```

Листинг 1: Вывод в консоль

```
1 Main thread created
2 Child thread created
3 n = 1000
4 sum = 0.49974987493746836
5 Child thread finished
6 Main thread finished
```

Вывод: Реализовали класс - поток (наследовали Thread). Реализовал вычисление в отдельном потоке (переопределили метод run, а вызывали, используя метод start). Присоединил поток к основному потоку, используя метод join, для того, чтобы основной поток не завершался, пока не выполнится дочерний.

Список использованных источников:

1. Thread в Java: Часть I - потоки [Электронный ресурс] - Режим доступа: <https://javarush.ru/groups/posts/2047-threadom-java-ne-isportishjh--chastjh-i---potoki>. Дата доступа: 15.02.2022.
2. Метод Thread.join() [Электронный ресурс] - Режим доступа: <https://javarush.ru/groups/posts/1993-mnogopotchnostjh-cto-delajut-metodih-klassa-thread>. Дата доступа: 15.02.2022.
3. Как создать исполняемый jar файл в IntelliJ IDEA - YouTube [Электронный ресурс] - Режим доступа: https://www.youtube.com/watch?v=tA8rEz_xFrQ. Дата доступа: 01.05.2022.
4. Export JavaFX 11, 15 or 17 projects into an executable jar file with IntelliJ [2022] - YouTube [Электронный ресурс] - Режим доступа: <https://www.youtube.com/watch?v=F8ahBtXkQzU>. Дата доступа: 01.05.2022.