Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Пенза 2022

Выполнили:

студенты группы 19ВВ2

Макаров Д.Д.

Лобанов Д.В.

Приняли:

Карамышева Н.С

Юрова О.В.

ОТЧЕТ

по лабораторной работе №5

по курсу «Программирование на языке Java»

на тему «Многопоточность в Java»

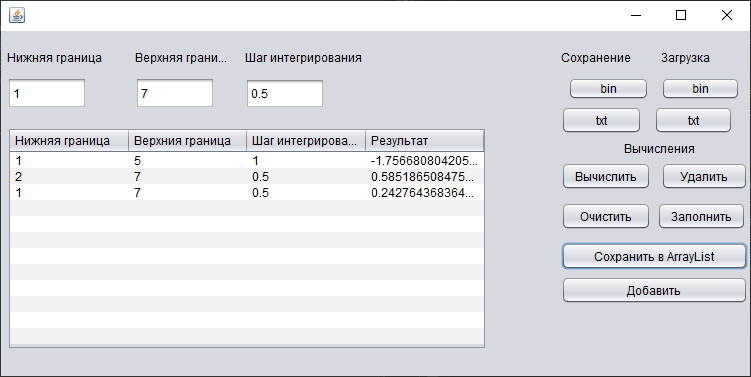
**Цель работы:**

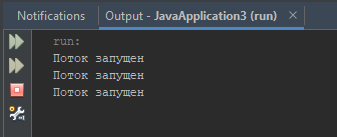
Научиться создавать многопоточные приложения c использованием стандартных средств языка Java.

**Задание на лабораторную работу:**

Модифицировать приложение из предыдущей лабораторной работы, реализовав вычисление определенного интеграла в нескольких дополнительных потоках (число потоков определяется номером варианта), снимая нагрузку с основного потока и предотвращая "подвисание" графического интерфейса. Варианты с номерами до 5 включительно реализуют многопоточность путем наследования от класса Thread, остальные реализуют интерфейс Runnable.

**Результат работы:**

****

****

**Вывод:** Мы научились создавать многопоточные приложения c использованием стандартных средств языка Java.

**Листинг:**

**ThreadIntegral.java**

package my.numberaddition;

public class ThreadIntegral extends Thread// implements Runnable

{

private Double dwBottom;

private Double dwHigh;

private Double dwStep;

private Double dwResult;

private int row;

@Override

public void run() {

System.out.print("Поток запущен\n");

while (dwBottom < dwHigh) {

Double temp = 0.0;

temp = Math.cos(Math.pow(dwBottom, 2)) / 2;

dwBottom += dwStep;

temp += Math.cos(Math.pow(dwBottom, 2)) / 2;

dwResult += temp \* dwStep;

}

}

ThreadIntegral() {

dwBottom = 0.0;

dwHigh = 0.0;

dwStep = 0.0;

dwResult = 0.0;

}

ThreadIntegral(Double dwBottom, Double dwHigh, Double dwStep, int row) {

this.dwBottom = dwBottom;

this.dwHigh = dwHigh;

this.dwStep = dwStep;

this.dwResult = 0.0;

this.row = row;

}

public Double GetResult() {

return dwResult;

}

}

**ContactEditorUI.java**

private void jButton3MouseClicked(java.awt.event.MouseEvent evt) throws InterruptedException

{

ThreadIntegral[] Thread = new ThreadIntegral[jTable2.getRowCount()];

for(int i = 0; i < jTable2.getRowCount(); i++)

{

if (jTable2.getValueAt(i, 1)!=null)

{

Thread[i] = new ThreadIntegral(

Double.parseDouble((String) jTable2.getValueAt(i, 0)),

Double.parseDouble((String) jTable2.getValueAt(i, 1)),

Double.parseDouble((String) jTable2.getValueAt(i, 2)),

i

);

Thread[i].start();

}

}

for(int i = 0; i < jTable2.getRowCount(); i++)

{

if (jTable2.getValueAt(i, 1)!=null)

{

Thread[i].join();

jTable2.setValueAt(Thread[i].GetResult().toString(), i, 3);

}

}

private void jButton11MouseClicked(java.awt.event.MouseEvent evt) {

for(int i = 0; i < jTable2.getRowCount(); i++)

{

if (jTable2.getValueAt(i, 1)!=null)

{

RecIntegral RecIntCapacitor = new RecIntegral();

RecIntCapacitor.SetBottom(Double.parseDouble((String)jTable2.getValueAt(i, 0)));

RecIntCapacitor.SetHigh(Double.parseDouble((String)jTable2.getValueAt(i, 1)));

RecIntCapacitor.SetStep(Double.parseDouble((String)jTable2.getValueAt(i, 2)));

RecIntCapacitor.SetResult((String)jTable2.getValueAt(i, 3));

Array.add(RecIntCapacitor);

}

}

}