Пензенский государственный университет

Факультет вычислительной техники

Кафедра «Вычислительная техника»

Отчет о лабораторной работе № 5

по дисциплине «Программирование на языке JAVA»

Вариант № 4

Выполнили: ст-ты гр. 19ВВ1

Васильев Л.Р

Проверили:

Юрова О.В.

Карамышева Н.С.

2022

**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.

**Решение:**

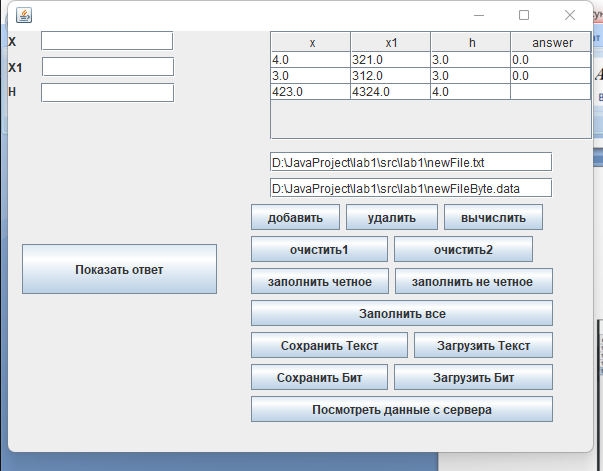


Рисунок 1 - Загрузка ЦП на 100% (5 потоков на 4 ядра)

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

**import java.io.OutputStreamWriter;**

**import java.net.Socket;**

**import java.util.logging.Level;**

**import java.util.logging.Logger;**

**import javax.swing.JOptionPane;**

**public class MyThread extends Thread{**

**RecIntegrall rec;**

**private float x = 0;**

**private float x1 = 0;**

**private float h = 0;**

**private int i = 0;**

**public MyThread(float X, float X1, float H, int I, RecIntegrall REC){**

**x = X;**

**x1 = X1;**

**h = H;**

**i = I;**

**rec = REC;**

**}**

**@Override**

**public void run(){**

**try {**

**sendRequest(x, x1, h);**

**} catch (IOException ex) {**

**Logger.getLogger(MyThread.class.getName()).log(Level.SEVERE, null, ex);**

**}**

**}**

**private void sendRequest(float x, float x1, float h) throws IOException {**

**Socket client = new Socket("localhost", 25225);**

**BufferedReader br = new BufferedReader(new InputStreamReader(client.getInputStream()));**

**BufferedWriter bw = new BufferedWriter(new OutputStreamWriter(client.getOutputStream()));**

**String sb = "";**

**sb += Float.toString(x) + ' ' + Float.toString(x1) + ' ' + Float.toString(h);**

**bw.write(sb);**

**bw.newLine();**

**bw.flush();**

**String answer = br.readLine();**

**if(i % 2 == 0){**

**rec.addArr(new DataLab(x, x1, h, Float.parseFloat(answer)));**

**}else{**

**rec.addArr1(new DataLab(x, x1, h, Float.parseFloat(answer)));**

**}**

**br.close();**

**bw.close();**

**}**

**}**

**private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {**

**MyThread[] thr = new MyThread[4];**

**DataLab data = null;**

**DefaultTableModel dtm = (DefaultTableModel) jTable1.getModel();**

**int nRow = dtm.getRowCount();**

**float[] answer = new float[nRow];**

**float x = 0;**

**float x1 = 0;**

**float h = 0;**

**int n = 0;**

**int n1 = 0;**

**rec.clearArr1();**

**rec.clearArr();**

**for (int i = 0 ; i < nRow ; i++){**

**x = Float.parseFloat((String) dtm.getValueAt(i, 0));**

**x1 = Float.parseFloat((String) dtm.getValueAt(i, 1));**

**h = Float.parseFloat((String) dtm.getValueAt(i, 2));**

**data = new DataLab(x, x1, h, 0);**

**if(i % 5 == 0){**

**try {**

**sendRequest(x, x1, h, i);**

**} catch (IOException ex) {**

**Logger.getLogger(lab2.class.getName()).log(Level.SEVERE, null, ex);**

**}**

**}else if(i % 5 == 1){**

**if(i % 2 == 0){**

**thr[0] = new MyThread(x, x1, h, i,rec);**

**thr[0].start();**

**}else{**

**thr[0] = new MyThread(x, x1, h, i,rec);**

**thr[0].start();**

**}**

**}else if(i % 5 == 2){**

**if(i % 2 == 0){**

**thr[1] = new MyThread(x, x1, h, i,rec);**

**thr[1].start();**

**}else{**

**thr[1] = new MyThread(x, x1, h, i,rec);**

**thr[1].start();**

**}**

**}else if(i % 5 == 3){**

**if(i % 2 == 0){**

**thr[2] = new MyThread(x, x1, h, i,rec);**

**thr[2].start();**

**}else{**

**thr[2] = new MyThread(x, x1, h, i,rec);**

**thr[2].start();**

**}**

**}else if(i % 5 == 4){**

**if(i % 2 == 0){**

**thr[3] = new MyThread(x, x1, h, i,rec);**

**thr[3].start();**

**}else{**

**thr[3] = new MyThread(x, x1, h, i,rec);**

**thr[3].start();**

**}**

**}**

**}**

**}**