Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №5

по курсу “Программирование на языке Java”

на тему: “Многопоточность в Java”

Вариант 2

Выполнили студенты группы 21ВВП1

Кривенкова В.С.

Сагателов А.К.

Митрошин А.Д.

Приняли:

к.т.н., доцент Юрова О.В.,

к.т.н., доцент Карамышева Н.С.

Пенза 2024

**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. Оформление лабораторной работы должно быть выполнено в соответствии с требованиями, приведенными в Приложении 2.

**Ход работы:**

Создали новый класс JThread, унаследованный от класса Thread.

class JThread extends Thread{

private double a;

private double b;

private double n;

private double result = 0.0;

JThread(double a, double b, double n){

this.a = a;

this.b = b;

this.n = n;

}

public double getResult () {

return result;

}

@Override

public void run(){

double dx = (b-a)/n;

for (int i = 1; i < dx; i++){

if(!((a+n)>b)) {

result+= (Math.sin(a+n)+Math.sin(a))\*n/2;

a += n;

}

else result+= (Math.sin(a)+Math.sin(b))\*(b-a)/2;

}

}

}

Изменили метод-обработчик кнопки «Вычислить»

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

DefaultTableModel tableModel = (DefaultTableModel) jTable1.getModel();

int row = jTable1.getSelectedRow();

RecIntegral item = ri.get(row);

if(row != -1){

double a = item.getA();

double b = item.getB();

double n = item.getN();

JThread Thread1 = new JThread(a, (b-a)/2, n);

JThread Thread2 = new JThread((b-a)/2, b, n);

Thread1.start();

Thread2.start();

try {

Thread1.join();

Thread2.join();

} catch (InterruptedException ex) {

JOptionPane.showMessageDialog(null, ex.getMessage());

}

double result = Thread1.getResult() + Thread2.getResult();

item.setResult(result);

tableModel.setValueAt(result, row, 3);

}

}

**Листинг программы:**

/\*

\* Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license

\* Click nbfs://nbhost/SystemFileSystem/Templates/GUIForms/JFrame.java to edit this template

\*/

import javax.swing.table.DefaultTableModel;

import java.util.LinkedList;

import javax.swing.JOptionPane;

import java.io.\*;

import java.util.Scanner;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.swing.JFileChooser;

/\*\*

\*

\* @author Artyom

\*/

public class NewJFrame extends javax.swing.JFrame {

/\*\*

\* Creates new form NewJFrame

\*/

public NewJFrame() {

initComponents();

}

/\*\*

\* This method is called from within the constructor to initialize the form.

\* WARNING: Do NOT modify this code. The content of this method is always

\* regenerated by the Form Editor.

\*/

@SuppressWarnings("unchecked")

// <editor-fold defaultstate="collapsed" desc="Generated Code">

private void initComponents() {

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jLabel1 = new javax.swing.JLabel();

jTextField1 = new javax.swing.JTextField();

jLabel2 = new javax.swing.JLabel();

jTextField2 = new javax.swing.JTextField();

jLabel3 = new javax.swing.JLabel();

jTextField3 = new javax.swing.JTextField();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jButton4 = new javax.swing.JButton();

jButton5 = new javax.swing.JButton();

jButton8 = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

jButton7 = new javax.swing.JButton();

jButton9 = new javax.swing.JButton();

setDefaultCloseOperation(javax.swing.WindowConstants.EXIT\_ON\_CLOSE);

setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

setResizable(false);

jTable1.setModel(new javax.swing.table.DefaultTableModel(

new Object [][] {

},

new String [] {

"Нижняя граница", "Верхняя граница", "Шаг", "Результат"

}

) {

boolean[] canEdit = new boolean [] {

false, false, false, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

jScrollPane1.setViewportView(jTable1);

jLabel1.setText("Нижняя граница");

jLabel1.setFocusCycleRoot(true);

jLabel1.setFocusable(false);

jTextField1.setActionCommand("<не установлено>");

jTextField1.setCursor(new java.awt.Cursor(java.awt.Cursor.TEXT\_CURSOR));

jLabel2.setText("Верхняя граница");

jLabel2.setFocusCycleRoot(true);

jLabel2.setFocusable(false);

jTextField2.setActionCommand("<не установлено>");

jTextField2.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jTextField2.setFocusCycleRoot(true);

jTextField2.setNextFocusableComponent(jTextField3);

jLabel3.setText("Шаг");

jLabel3.setFocusCycleRoot(true);

jLabel3.setFocusable(false);

jTextField3.setActionCommand("<не установлено>");

jTextField3.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jTextField3.setFocusCycleRoot(true);

jButton1.setText("Добавить");

jButton1.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton1.setFocusCycleRoot(true);

jButton1.setFocusable(false);

jButton1.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton1ActionPerformed(evt);

}

});

jButton2.setText("Удалить");

jButton2.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton2.setFocusCycleRoot(true);

jButton2.setFocusable(false);

jButton2.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton2ActionPerformed(evt);

}

});

jButton3.setText("Вычислить");

jButton3.setFocusCycleRoot(true);

jButton3.setFocusable(false);

jButton3.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton3ActionPerformed(evt);

}

});

jButton4.setText("Заполнить");

jButton4.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton4.setFocusCycleRoot(true);

jButton4.setFocusable(false);

jButton4.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton4ActionPerformed(evt);

}

});

jButton5.setText("Очистить");

jButton5.setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT\_CURSOR));

jButton5.setFocusCycleRoot(true);

jButton5.setFocusable(false);

jButton5.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton5ActionPerformed(evt);

}

});

jButton8.setText("Сохр. в байт");

jButton8.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton8ActionPerformed(evt);

}

});

jButton6.setText("Сохр. в текст");

jButton6.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton6ActionPerformed(evt);

}

});

jButton7.setText("Загр. текст");

jButton7.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton7ActionPerformed(evt);

}

});

jButton9.setText("Загр. байт");

jButton9.addActionListener(new java.awt.event.ActionListener() {

public void actionPerformed(java.awt.event.ActionEvent evt) {

jButton9ActionPerformed(evt);

}

});

javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(jButton3, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel3, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel2, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jLabel1, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton2, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jButton1, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, 97, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 394, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING, false)

.addComponent(jButton5, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton4, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton8, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton6, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton7, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton9, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

.addContainerGap(javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE))

);

layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(layout.createSequentialGroup()

.addComponent(jButton4)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton5)

.addGap(4, 4, 4)

.addComponent(jButton6)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton7)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton8)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton9))

.addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 237, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(layout.createSequentialGroup()

.addComponent(jLabel1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jTextField1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jTextField2, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jLabel3)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jTextField3, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton1)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton2)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)

.addComponent(jButton3))))

.addContainerGap())

);

pack();

}// </editor-fold>

static class RecIntegral implements Serializable{

private double a;

private double b;

private double n;

private double result = 0.0;

RecIntegral()

{

a = 0.0;

b = 0.0;

n = 0.0;

}

RecIntegral(double a, double b, double n)

{

this.a = a;

this.b = b;

this.n = n;

}

RecIntegral(double a, double b, double n, double result)

{

this.a = a;

this.b = b;

this.n = n;

this.result = result;

}

double getA()

{

return a;

}

void setA(double a) throws RecIntegralException

{

this.a = a;

}

double getB()

{

return b;

}

void setB(double b) throws RecIntegralException

{

this.b = b;

}

double getN()

{

return n;

}

void setN(double n) throws RecIntegralException

{

this.n = n;

}

double getResult()

{

return result;

}

void setResult(double result)

{

this.result = result;

}

}

class RecIntegralException extends Exception{

public RecIntegralException(String message){

super(message);

}

}

class JThread extends Thread{

private double a;

private double b;

private double n;

private double result = 0.0;

JThread(double a, double b, double n){

this.a = a;

this.b = b;

this.n = n;

}

public double getResult () {

return result;

}

@Override

public void run(){

double dx = (b-a)/n;

for (int i = 1; i < dx; i++){

if(!((a+n)>b)) {

result+= (Math.sin(a+n)+Math.sin(a))\*n/2;

a += n;

}

else result+= (Math.sin(a)+Math.sin(b))\*(b-a)/2;

}

}

}

LinkedList<RecIntegral> ri = new LinkedList<>();

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

DefaultTableModel tableModel = (DefaultTableModel) jTable1.getModel();

int row = jTable1.getSelectedRow();

RecIntegral item = ri.get(row);

if(row != -1){

double a = item.getA();

double b = item.getB();

double n = item.getN();

JThread Thread1 = new JThread(a, (b-a)/2, n);

JThread Thread2 = new JThread((b-a)/2, b, n);

Thread1.start();

Thread2.start();

try {

Thread1.join();

Thread2.join();

} catch (InterruptedException ex) {

JOptionPane.showMessageDialog(null, ex.getMessage());

}

double result = Thread1.getResult() + Thread2.getResult();

item.setResult(result);

tableModel.setValueAt(result, row, 3);

}

}

private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel tableModel = (DefaultTableModel)jTable1.getModel();

ri.remove(jTable1.getSelectedRow());

tableModel.removeRow(jTable1.getSelectedRow());

}

private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {

try {

DefaultTableModel tableModel = (DefaultTableModel)jTable1.getModel();

if (Double.parseDouble(jTextField1.getText()) < 0.000001 || Double.parseDouble(jTextField1.getText()) > 1000000) {

throw new RecIntegralException("Значение нижней границы не может превышать диапазон от 0.000001 до 1000000");

}

else if (Double.parseDouble(jTextField2.getText()) < 0.000001 || Double.parseDouble(jTextField2.getText()) > 1000000) {

throw new RecIntegralException("Значение верхней границы не может превышать диапазон от 0.000001 до 1000000");

}

else if (Double.parseDouble(jTextField1.getText()) >= Double.parseDouble(jTextField2.getText())){

throw new RecIntegralException("Значение нижней границы не может быть больше либо равно, чем значение верхней границы");

}

else if (Double.parseDouble(jTextField3.getText()) < 0.000001 || Double.parseDouble(jTextField3.getText()) > 1000000){

throw new RecIntegralException("Значение шага не может превышать диапазон от 0.000001 до 1000000");

}

else if (Double.parseDouble(jTextField3.getText()) > Double.parseDouble(jTextField2.getText())){

throw new RecIntegralException("Значение шага не может быть больше, чем значение верхней границы");

}

ri.add(new RecIntegral(Double.parseDouble(jTextField1.getText()), Double.parseDouble(jTextField2.getText()), Double.parseDouble(jTextField3.getText())));

tableModel.addRow(new Object[]{jTextField1.getText(),jTextField2.getText(),jTextField3.getText()});

jTextField1.setText("");

jTextField2.setText("");

jTextField3.setText("");

} catch(RecIntegralException ex) {

JOptionPane.showMessageDialog(null, ex.getMessage());

}

}

private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel tableModel = (DefaultTableModel)jTable1.getModel();

for (RecIntegral item : ri)

{

tableModel.addRow(new Object[]{item.getA(),item.getB(),item.getN(),item.getResult()});

}

}

private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel tableModel = (DefaultTableModel)jTable1.getModel();

int rows = tableModel.getRowCount();

while (rows != 0)

{

tableModel.removeRow(0);

rows--;

}

}

/// Сохранение в байты

private void jButton8ActionPerformed(java.awt.event.ActionEvent evt) {

JFileChooser chooser = new JFileChooser();

ObjectOutputStream out;

int wnd = chooser.showSaveDialog(NewJFrame.this);

if (wnd == JFileChooser.APPROVE\_OPTION){

try {

out = new ObjectOutputStream(new BufferedOutputStream(new FileOutputStream(chooser.getSelectedFile())));

out.writeObject(ri);

out.close();

} catch (IOException ex)

{

JOptionPane.showMessageDialog(null, ex.getMessage());

}

}

}

/// Сохранение в текст

private void jButton6ActionPerformed(java.awt.event.ActionEvent evt) {

JFileChooser chooser = new JFileChooser();

int wnd = chooser.showSaveDialog(NewJFrame.this);

if (wnd == JFileChooser.APPROVE\_OPTION){

try {

FileWriter fw = new FileWriter(chooser.getSelectedFile());

for (RecIntegral item : ri) {

if (ri.getLast() != item)

{

fw.write(String.valueOf(item.getA()) + " " + String.valueOf(item.getB()) + " "

+ String.valueOf(item.getN()) + " " + String.valueOf(item.getResult()) + "\n");

}

else

{

fw.write(String.valueOf(item.getA()) + " " + String.valueOf(item.getB()) + " "

+ String.valueOf(item.getN()) + " " + String.valueOf(item.getResult()));

}

}

fw.close();

} catch (IOException ex)

{

JOptionPane.showMessageDialog(null, ex.getMessage());

}

}

}

/// Загрузка из текста

private void jButton7ActionPerformed(java.awt.event.ActionEvent evt) {

JFileChooser chooser = new JFileChooser("C:\\Users\\iljak\\Desktop");

int wnd = chooser.showOpenDialog(NewJFrame.this);

if (wnd == JFileChooser.APPROVE\_OPTION){

try {

FileReader fr = new FileReader(chooser.getSelectedFile());

Scanner scan = new Scanner(fr);

while (scan.hasNextLine())

{

ri.add(new RecIntegral(Double.parseDouble(scan.next()), Double.parseDouble(scan.next()),

Double.parseDouble(scan.next()), Double.parseDouble(scan.next())));

}

jButton5ActionPerformed(evt);

jButton4ActionPerformed(evt);

fr.close();

} catch (IOException ex)

{

JOptionPane.showMessageDialog(null, ex.getMessage());

}

}

}

/// Загрузка байт

private void jButton9ActionPerformed(java.awt.event.ActionEvent evt) {

JFileChooser chooser = new JFileChooser("C:\\Users\\iljak\\Desktop");

ObjectInputStream oit;

int wnd = chooser.showOpenDialog(NewJFrame.this);

if (wnd == JFileChooser.APPROVE\_OPTION){

try {

oit = new ObjectInputStream(new BufferedInputStream(new FileInputStream(chooser.getSelectedFile())));

try {

ri.addAll((LinkedList<RecIntegral>)oit.readObject());

jButton5ActionPerformed(evt);

jButton4ActionPerformed(evt);

} catch (ClassNotFoundException ex) {

JOptionPane.showMessageDialog(null, ex.getMessage());

}

oit.close();

} catch (IOException ex) {

JOptionPane.showMessageDialog(null, ex.getMessage());

}

}

}

/\*\*

\* @param args the command line arguments

\*/

public static void main(String args[]) {

/\* Set the Nimbus look and feel \*/

//<editor-fold defaultstate="collapsed" desc=" Look and feel setting code (optional) ">

/\* If Nimbus (introduced in Java SE 6) is not available, stay with the default look and feel.

\* For details see http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html

\*/

try {

for (javax.swing.UIManager.LookAndFeelInfo info : javax.swing.UIManager.getInstalledLookAndFeels()) {

if ("Nimbus".equals(info.getName())) {

javax.swing.UIManager.setLookAndFeel(info.getClassName());

break;

}

}

} catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(NewJFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

}

//</editor-fold>

/\* Create and display the form \*/

java.awt.EventQueue.invokeLater(new Runnable() {

public void run() {

new NewJFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton jButton1;

private javax.swing.JButton jButton2;

private javax.swing.JButton jButton3;

private javax.swing.JButton jButton4;

private javax.swing.JButton jButton5;

private javax.swing.JButton jButton6;

private javax.swing.JButton jButton7;

private javax.swing.JButton jButton8;

private javax.swing.JButton jButton9;

private javax.swing.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JScrollPane jScrollPane1;

private javax.swing.JTable jTable1;

private javax.swing.JTextField jTextField1;

private javax.swing.JTextField jTextField2;

private javax.swing.JTextField jTextField3;

// End of variables declaration

}

**Результат работы программы**

Результат работы программы приведён на рисунке 1.

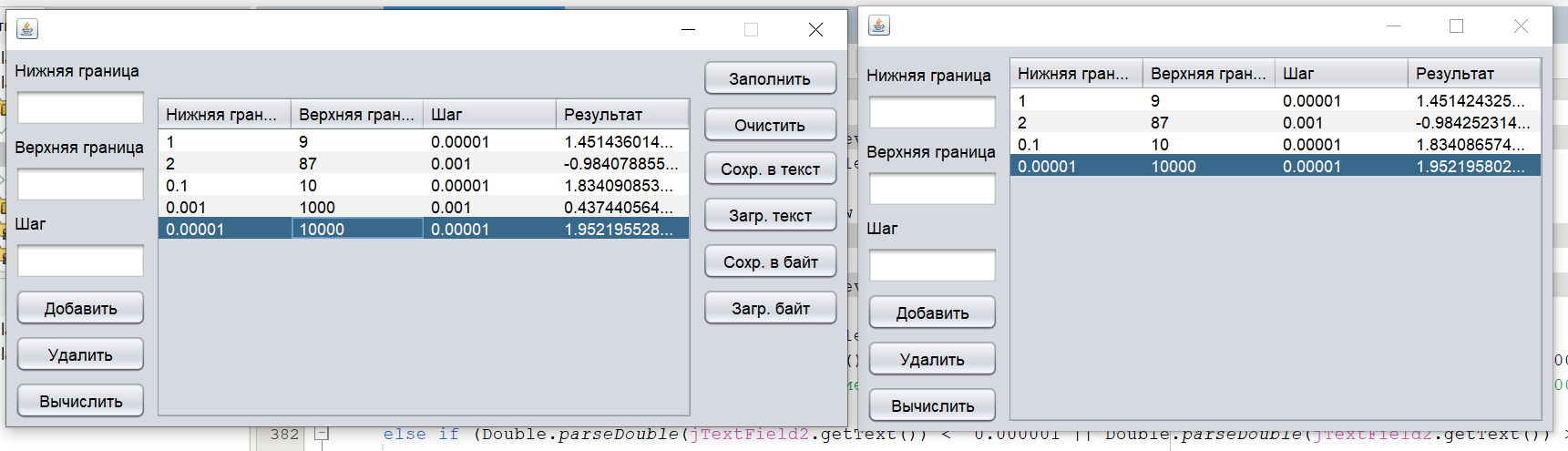


Рисунок 1 – Сравнение результата с результатом из первой л.р.

Вывод: в ходе выполнения данной лабораторной работы мы научились создавать многопоточные приложения c использованием стандартных средств языка Java.