Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №5

по курсу «Разработка кроссплатформенных приложений»

на тему «Многопоточность в Java»

**Выполнили студенты группы 21ввв1:**

Гришин С.В.

Кузнецов А.С.

**Приняли**

Юрова О.В.

Карамышева Н.С.

Пенза 2024

### Цель работы

научиться создавать многопоточные приложения c использованием стандартных средств языка Java.

### Лабораторное задание

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

**Листинг:**

import javax.swing.table.DefaultTableModel;

import java.util.Collection;

import java.util.LinkedList;

import java.lang.Math;

import javax.swing.JOptionPane;

import com.formdev.flatlaf.FlatDarkLaf;

import javax.swing.UIManager;

import javax.swing.\*;

import java.io.\*;

import javax.swing.filechooser.FileNameExtensionFilter;

import java.util.Scanner;

/\*\*

\*

\* @author RBDstudio

\*/

//public class RecIntegral

public class JFrame extends javax.swing.JFrame {

public JFrame() {

initComponents();

}

@SuppressWarnings("unchecked")

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

private void initComponents() {

jFileChooser1 = new javax.swing.JFileChooser();

lower\_bound\_ent = new javax.swing.JTextField();

step\_ent = new javax.swing.JTextField();

upper\_bound\_ent = new javax.swing.JTextField();

Add\_button = new javax.swing.JButton();

UBtext = new javax.swing.JLabel();

DBtext = new javax.swing.JLabel();

Stext = new javax.swing.JLabel();

Logo = new javax.swing.JLabel();

MainTable = new javax.swing.JTabbedPane();

CurrentTable = new javax.swing.JScrollPane();

CurTable = new javax.swing.JTable();

ConteinerTable = new javax.swing.JScrollPane();

ConTable = new javax.swing.JTable();

ActionMenu = new javax.swing.JTabbedPane();

CalTable = new javax.swing.JScrollPane();

jPanel1 = new javax.swing.JPanel();

Calcul\_button = new javax.swing.JButton();

DeleteMain\_Button = new javax.swing.JButton();

SaveConteiner\_Button = new javax.swing.JButton();

DeleteMain\_Button1 = new javax.swing.JButton();

SaveTable = new javax.swing.JScrollPane();

jPanel2 = new javax.swing.JPanel();

Return\_Button = new javax.swing.JButton();

DeteleCont\_button = new javax.swing.JButton();

CleanCont\_Button = new javax.swing.JButton();

jMenuBar1 = new javax.swing.JMenuBar();

jMenu1 = new javax.swing.JMenu();

jMenu2 = new javax.swing.JMenu();

jMenuItem1 = new javax.swing.JMenuItem();

jMenuItem2 = new javax.swing.JMenuItem();

jMenu3 = new javax.swing.JMenu();

jMenuItem3 = new javax.swing.JMenuItem();

jMenuItem4 = new javax.swing.JMenuItem();

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

setAutoRequestFocus(false);

setBackground(new java.awt.Color(51, 51, 51));

setResizable(false);

lower\_bound\_ent.setText("0");

lower\_bound\_ent.setToolTipText("введите нижнюю границу");

lower\_bound\_ent.addActionListener(new java.awt.event.ActionListener() {

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

lower\_bound\_entActionPerformed(evt);

}

});

step\_ent.setText("0.001");

step\_ent.setToolTipText("введите шаг");

upper\_bound\_ent.setText("1");

upper\_bound\_ent.setToolTipText("введите верхнюю границу");

upper\_bound\_ent.addActionListener(new java.awt.event.ActionListener() {

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

upper\_bound\_entActionPerformed(evt);

}

});

Add\_button.setText("добавить");

Add\_button.setToolTipText("добавляет введёные значения в таблицу");

Add\_button.addActionListener(new java.awt.event.ActionListener() {

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

Add\_buttonActionPerformed(evt);

}

});

UBtext.setText("верхняя граница");

DBtext.setText("нижняя граница");

Stext.setText(" шаг");

Stext.setVerticalTextPosition(javax.swing.SwingConstants.TOP);

Logo.setIcon(new javax.swing.ImageIcon(getClass().getResource("/logo.jpg"))); // NOI18N

Logo.setText("jLabel4");

MainTable.setBorder(javax.swing.BorderFactory.createCompoundBorder());

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

new Object [][] {

},

new String [] {

"нижняя гр.", "верхняя гр.", "шаг", "результат"

}

) {

boolean[] canEdit = new boolean [] {

true, true, true, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

CurTable.setSelectionMode(javax.swing.ListSelectionModel.SINGLE\_SELECTION);

CurrentTable.setViewportView(CurTable);

MainTable.addTab("главная", CurrentTable);

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

new Object [][] {

},

new String [] {

"нижняя гр.", "верхняя гр.", "шаг", "результат"

}

) {

boolean[] canEdit = new boolean [] {

false, false, false, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

ConTable.setSelectionMode(javax.swing.ListSelectionModel.SINGLE\_SELECTION);

ConTable.setShowGrid(false);

ConteinerTable.setViewportView(ConTable);

MainTable.addTab("контейнер", ConteinerTable);

ActionMenu.setToolTipText("выбор режима");

CalTable.setHorizontalScrollBarPolicy(javax.swing.ScrollPaneConstants.HORIZONTAL\_SCROLLBAR\_NEVER);

Calcul\_button.setText("вычислить");

Calcul\_button.setToolTipText("вычисляет значение выбранной строки");

Calcul\_button.addActionListener(new java.awt.event.ActionListener() {

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

Calcul\_buttonActionPerformed(evt);

}

});

DeleteMain\_Button.setText("удалить");

DeleteMain\_Button.setToolTipText("удаляет выбранную строку");

DeleteMain\_Button.addActionListener(new java.awt.event.ActionListener() {

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

DeleteMain\_ButtonActionPerformed(evt);

}

});

SaveConteiner\_Button.setText("записать в контейнер");

SaveConteiner\_Button.setToolTipText("помещает выбранную строку в контейнер");

SaveConteiner\_Button.addActionListener(new java.awt.event.ActionListener() {

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

SaveConteiner\_ButtonActionPerformed(evt);

}

});

DeleteMain\_Button1.setText("очистить");

DeleteMain\_Button1.setToolTipText("удаляет выбранную строку");

DeleteMain\_Button1.addActionListener(new java.awt.event.ActionListener() {

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

DeleteMain\_Button1ActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel1Layout = new javax.swing.GroupLayout(jPanel1);

jPanel1.setLayout(jPanel1Layout);

jPanel1Layout.setHorizontalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(DeleteMain\_Button1, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(Calcul\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(DeleteMain\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(SaveConteiner\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addContainerGap(19, Short.MAX\_VALUE))

);

jPanel1Layout.setVerticalGroup(

jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(Calcul\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 49, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(SaveConteiner\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(DeleteMain\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(DeleteMain\_Button1, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

);

CalTable.setViewportView(jPanel1);

ActionMenu.addTab("главная", CalTable);

SaveTable.setHorizontalScrollBarPolicy(javax.swing.ScrollPaneConstants.HORIZONTAL\_SCROLLBAR\_NEVER);

Return\_Button.setText("вернуть");

Return\_Button.setToolTipText("записывает выбранное значение в основную таблицу");

Return\_Button.addActionListener(new java.awt.event.ActionListener() {

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

Return\_ButtonActionPerformed(evt);

}

});

DeteleCont\_button.setText("удалить");

DeteleCont\_button.setToolTipText("удаляет выбранное значение из контейнера");

DeteleCont\_button.addActionListener(new java.awt.event.ActionListener() {

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

DeteleCont\_buttonActionPerformed(evt);

}

});

CleanCont\_Button.setText("очистить");

CleanCont\_Button.setToolTipText("очищает контейнер");

CleanCont\_Button.addActionListener(new java.awt.event.ActionListener() {

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

CleanCont\_ButtonActionPerformed(evt);

}

});

javax.swing.GroupLayout jPanel2Layout = new javax.swing.GroupLayout(jPanel2);

jPanel2.setLayout(jPanel2Layout);

jPanel2Layout.setHorizontalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

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

.addGroup(jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addComponent(Return\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(DeteleCont\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(CleanCont\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 153, javax.swing.GroupLayout.PREFERRED\_SIZE))

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

);

jPanel2Layout.setVerticalGroup(

jPanel2Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)

.addGroup(jPanel2Layout.createSequentialGroup()

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

.addComponent(Return\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 49, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(DeteleCont\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(CleanCont\_Button, javax.swing.GroupLayout.PREFERRED\_SIZE, 47, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

);

SaveTable.setViewportView(jPanel2);

ActionMenu.addTab("контейнер", SaveTable);

jMenu1.setText("Файл");

jMenu2.setText("Сохранение");

jMenuItem1.setText("текстовой вид");

jMenuItem1.addActionListener(new java.awt.event.ActionListener() {

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

jMenuItem1ActionPerformed(evt);

}

});

jMenu2.add(jMenuItem1);

jMenuItem2.setText("двоичный вид");

jMenuItem2.addActionListener(new java.awt.event.ActionListener() {

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

jMenuItem2ActionPerformed(evt);

}

});

jMenu2.add(jMenuItem2);

jMenu1.add(jMenu2);

jMenu3.setText("Загрузка");

jMenuItem3.setText("текстовой вид");

jMenuItem3.addActionListener(new java.awt.event.ActionListener() {

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

jMenuItem3ActionPerformed(evt);

}

});

jMenu3.add(jMenuItem3);

jMenuItem4.setText("двоичный вид");

jMenuItem4.addActionListener(new java.awt.event.ActionListener() {

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

jMenuItem4ActionPerformed(evt);

}

});

jMenu3.add(jMenuItem4);

jMenu1.add(jMenu3);

jMenuBar1.add(jMenu1);

setJMenuBar(jMenuBar1);

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(ActionMenu, javax.swing.GroupLayout.PREFERRED\_SIZE, 180, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGroup(layout.createSequentialGroup()

.addGap(17, 17, 17)

.addComponent(Logo, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(18, 18, 18)

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

.addGroup(layout.createSequentialGroup()

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

.addGroup(layout.createSequentialGroup()

.addComponent(DBtext, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(UBtext, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(Stext, javax.swing.GroupLayout.PREFERRED\_SIZE, 67, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addComponent(lower\_bound\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(upper\_bound\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(step\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 100, javax.swing.GroupLayout.PREFERRED\_SIZE)))

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

.addComponent(Add\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 98, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addComponent(MainTable, javax.swing.GroupLayout.PREFERRED\_SIZE, 446, javax.swing.GroupLayout.PREFERRED\_SIZE))

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

);

layout.setVerticalGroup(

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

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, layout.createSequentialGroup()

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

.addGroup(layout.createSequentialGroup()

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

.addComponent(Logo, javax.swing.GroupLayout.PREFERRED\_SIZE, 150, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(ActionMenu, javax.swing.GroupLayout.PREFERRED\_SIZE, 228, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGroup(layout.createSequentialGroup()

.addGap(6, 6, 6)

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

.addComponent(Stext, javax.swing.GroupLayout.PREFERRED\_SIZE, 30, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

.addComponent(UBtext, javax.swing.GroupLayout.PREFERRED\_SIZE, 30, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(DBtext, javax.swing.GroupLayout.PREFERRED\_SIZE, 30, javax.swing.GroupLayout.PREFERRED\_SIZE)))

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

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

.addComponent(step\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(lower\_bound\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(upper\_bound\_ent, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(Add\_button, javax.swing.GroupLayout.PREFERRED\_SIZE, 31, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(MainTable, javax.swing.GroupLayout.PREFERRED\_SIZE, 304, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addContainerGap())

);

pack();

}// </editor-fold>

private void lower\_bound\_entActionPerformed(java.awt.event.ActionEvent evt) {

}

public class RecIntegral

{

public double Lower\_bound;

public double Upper\_bound;

public double Step;

public double Result;

public RecIntegral()

{

Lower\_bound = 0;

Upper\_bound = 0;

Step = 0;

Result = 0;

}

public RecIntegral(double Lower\_bound, double Upper\_bound, double Step, double Result)throws RecIntegralExcept

{

this.Lower\_bound = Lower\_bound;

this.Upper\_bound = Upper\_bound;

this.Step = Step;

this.Result = Result;

}

public void SetData(double Lower\_bound, double Upper\_bound, double Step, double Result)

{

this.Lower\_bound = Lower\_bound;

this.Upper\_bound = Upper\_bound;

this.Step = Step;

this.Result = Result;

}

}

class RecIntegralExcept extends Exception

{

private int Number;

public int getNumber(){return Number;}

public RecIntegralExcept(String message, int Number)

{

super(message);

this.Number = Number;

}

}

public class ThreadCalculateClass extends Thread

{

private double Upper\_bound, Lower\_bound, Step;

double Res = 0;

public ThreadCalculateClass(double Upper\_bound, double Lower\_bound, double Step)

{

this.Upper\_bound = Upper\_bound;

this.Lower\_bound = Lower\_bound;

this.Step = Step;

}

public void run()

{

for (double i = Lower\_bound; i < Upper\_bound; i=i+Step)

{

if(i+Step > Upper\_bound)

{

Step = Upper\_bound-i;

}

Res += Step \* (Math.tan(i) + Math.tan(i+Step)) / 2;

}

}

public double GetResult(){return Res;}

}

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

public double calculate(double Upper\_bound, double Lower\_bound, double Step)

{

double Res = 0;

for (double i = Lower\_bound; i < Upper\_bound; i=i+Step)

{

if(i+Step > Upper\_bound)

{

Step = Upper\_bound-i;

}

Res += Step \* (Math.tan(i) + Math.tan(i+Step)) / 2;

}

return Res;

}

public double MutliThreadCalculate(double Upper\_bound, double Lower\_bound, double Step, int ThreadCount)

{

double Res = 0;

ThreadCalculateClass[] threads = new ThreadCalculateClass[ThreadCount];

double step = (Upper\_bound - Lower\_bound)/ ThreadCount;

double b = 3;

// Создание и запуск потоков

for (int i = 0; i < ThreadCount; i++)

{

double lowerBound = i \* step;

double upper = Math.min((i + 1) \* step, Upper\_bound);

b = upper;

threads[i] = new ThreadCalculateClass(upper, lowerBound,Step);

threads[i].start();

}

// Ожидание завершения всех потоков

try

{

for (ThreadCalculateClass thread : threads) {

thread.join();

Res += thread.GetResult();

}

} catch(Exception e)

{

JOptionPane.showMessageDialog(null, "Ошибка при сохранении (" + e.getMessage() + ")");

}

return Res;

}

////////////////////////////////////////////////////////////////////////////////////////////////////

//ВЫЧИСЛЕНИЕ

private void Calcul\_buttonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

int selectedRow = CurTable.getSelectedRow();

double Lower\_bound = 0, Upper\_bound = 0,Step = 0, Res = 0;

if(selectedRow!=-1)

{

Lower\_bound = Double.parseDouble(myTable.getValueAt(selectedRow, 0).toString());

Upper\_bound = Double.parseDouble(myTable.getValueAt(selectedRow, 1).toString());

Step = Double.parseDouble(myTable.getValueAt(selectedRow, 2).toString());

}

int Tcount = (int)(Upper\_bound - Lower\_bound) / 4;

Res = MutliThreadCalculate(Upper\_bound, Lower\_bound, Step,4);

myTable.setValueAt(Res, selectedRow, 3);

}

//ДОБАВЛЕНИЕ

private void Add\_buttonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

myTable.insertRow(0, new Object[]{

lower\_bound\_ent.getText(),

upper\_bound\_ent.getText(),

step\_ent.getText(),

});

double Lower\_bound = 0, Upper\_bound = 0,Step = 0, Res = 0;

Lower\_bound = Double.parseDouble(myTable.getValueAt(0, 0).toString());

Upper\_bound = Double.parseDouble(myTable.getValueAt(0, 1).toString());

Step = Double.parseDouble(myTable.getValueAt(0, 2).toString());

Res = calculate(Upper\_bound, Lower\_bound, Step);

myTable.setValueAt(Res, 0, 3);

}

//УДАЛЕНИЕ

private void DeleteMain\_ButtonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

int selectedRow = CurTable.getSelectedRow();

if(selectedRow!=-1)

{

myTable.removeRow(selectedRow);

}

}

private void upper\_bound\_entActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

//ДОБАВЛЕНИЕ В КОНТЕЙНЕР

private void SaveConteiner\_ButtonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

int selectedRow = CurTable.getSelectedRow();

if(selectedRow!=-1)

{

double Lower\_bound = 0, Upper\_bound = 0,Step = 0, Res = 0;

Lower\_bound = Double.parseDouble(myTable.getValueAt(selectedRow, 0).toString());

Upper\_bound = Double.parseDouble(myTable.getValueAt(selectedRow, 1).toString());

Step = Double.parseDouble(myTable.getValueAt(selectedRow, 2).toString());

Res = Double.parseDouble(myTable.getValueAt(selectedRow, 3).toString());

try

{

if(Lower\_bound > 100 || Upper\_bound > 100 || Step > 100) throw new RecIntegralExcept("Одно из введённых занчений больше 100",1);

if(Lower\_bound < 0.0001 || Upper\_bound < 0.0001 || Step < 0.0001) throw new RecIntegralExcept("Одно из введённых занчений меньше 0.0001",1);

if(Lower\_bound >= Upper\_bound) throw new RecIntegralExcept("неверно введены пороги",2);

if((Upper\_bound - Lower\_bound) < Step) throw new RecIntegralExcept("неверно введён шаг",3);

ConteinerData.add(new RecIntegral(Lower\_bound,Upper\_bound,Step,Res));

RecIntegral ConData = ConteinerData.getLast();

DefaultTableModel ConteinerTable =(DefaultTableModel)ConTable.getModel();

ConteinerTable.insertRow(ConteinerTable.getRowCount(), new Object[]{

ConData.Lower\_bound,

ConData.Upper\_bound,

ConData.Step,

ConData.Result,

});

} catch(RecIntegralExcept e)

{

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

}

}

}

//ВОЗВРАТ ИЗ КОНТЕЙНЕРА

private void Return\_ButtonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel ConteinerTable =(DefaultTableModel)ConTable.getModel();

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

int selectedRow = ConTable.getSelectedRow();

//RecIntegral ConData = ConteinerData.get(selectedRow);

for(int i = 0; i < ConteinerData.size();i++ )

{

RecIntegral ConData = ConteinerData.get(i);

myTable.insertRow(0, new Object[]{

ConData.Lower\_bound,

ConData.Upper\_bound,

ConData.Step,

ConData.Result,

});

}

}

//УДАЛЕНИЕ ИЗ КОНТЕЙНЕРА

private void DeteleCont\_buttonActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel ConteinerTable =(DefaultTableModel)ConTable.getModel();

int selectedRow = ConTable.getSelectedRow();

if(selectedRow!=-1)

{

ConteinerData.remove(selectedRow);

ConteinerTable.removeRow(selectedRow);

}

}

//ОЧИСТКА КОНТЕЙНЕРА

private void CleanCont\_ButtonActionPerformed(java.awt.event.ActionEvent evt) {

ConteinerData.clear();

DefaultTableModel ConteinerTable =(DefaultTableModel)ConTable.getModel();

ConteinerTable.setRowCount(0);

}

//СОХРАНЕНИЕ ТЕКСТ

private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

int countRow = myTable.getRowCount();

int countColum = myTable.getColumnCount();

if (countRow > 0)

{

FileNameExtensionFilter filter = new FileNameExtensionFilter("txt", "txt", "text");

jFileChooser1.setFileFilter(filter);

jFileChooser1.showSaveDialog(DBtext);

FileWriter fileWriter = new FileWriter(jFileChooser1.getSelectedFile() + ".txt");

for(int i = 0; i < countRow; i++)

{

for(int j = 0; j < countColum; j++)

{

fileWriter.write(myTable.getValueAt(i, j).toString() + "\n");

}

}

fileWriter.close();

JOptionPane.showMessageDialog(null, "Успешно сохраненно");

}

else

{

JOptionPane.showMessageDialog(null, "Нельзя сохранить пустую таблицу");

}

}catch(Exception e)

{

JOptionPane.showMessageDialog(null, "Ошибка при сохранении (" + e.getMessage() + ")");

}

}

//СОХРАНЕНИЕ ДВОИЧНЫЙ

private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

if(myTable.getColumnCount()> 0)

{

jFileChooser1.showSaveDialog(DBtext);

FileOutputStream FOS = new FileOutputStream(jFileChooser1.getSelectedFile() + ".ser");

ObjectOutputStream OOS = new ObjectOutputStream(FOS);

int count = myTable.getRowCount();

OOS.writeObject(count);

for(int i = 0; i < count; i++)

{

OOS.writeObject(new RecIntegral(

Double.parseDouble(myTable.getValueAt(i, 0).toString()),

Double.parseDouble(myTable.getValueAt(i, 1).toString()),

Double.parseDouble(myTable.getValueAt(i, 2).toString()),

Double.parseDouble(myTable.getValueAt(i, 3).toString())

));

}

OOS.close();

JOptionPane.showMessageDialog(null, "Успешно сохраненно");

}

else

{

JOptionPane.showMessageDialog(null, "Нельзя сохранить пустую таблицу");

}

} catch(Exception e)

{

JOptionPane.showMessageDialog(null, "Ошибка при сохранении (" + e.getMessage() + ")");

}

}

//ЗАГРУЗКА ТЕКСТ

private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

FileNameExtensionFilter filter = new FileNameExtensionFilter("txt", "txt", "text");

jFileChooser1.setFileFilter(filter);

jFileChooser1.showOpenDialog(DBtext);

FileReader fileReader = new FileReader(jFileChooser1.getSelectedFile());

Scanner scan = new Scanner(fileReader);

myTable.setRowCount(0);

while (scan.hasNextLine())

{

myTable.insertRow(myTable.getRowCount(), new Object[]

{

scan.nextLine(),

scan.nextLine(),

scan.nextLine(),

scan.nextLine(),

});

}

}catch(Exception e)

{

JOptionPane.showMessageDialog(null, "Ошибка при загрузке (" + e.getMessage() + ")");

}

}

//ЗАГРУЗКА ДВОИЧНЫЙ

private void jMenuItem4ActionPerformed(java.awt.event.ActionEvent evt) {

try

{

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

FileNameExtensionFilter filter = new FileNameExtensionFilter("ser", "ser", "ser");

jFileChooser1.setFileFilter(filter);

jFileChooser1.showOpenDialog(DBtext);

FileInputStream FIS = new FileInputStream(jFileChooser1.getSelectedFile());

ObjectInputStream OIS = new ObjectInputStream(FIS);

int count = (int)OIS.readObject();

myTable.setRowCount(0);

for(int i = 0; i < count; i ++)

{

RecIntegral buf = (RecIntegral)OIS.readObject();

myTable.insertRow(myTable.getRowCount(), new Object[]

{

buf.Lower\_bound,

buf.Upper\_bound,

buf.Step,

buf.Result

});

}

OIS.close();

} catch(Exception e)

{

JOptionPane.showMessageDialog(null, "Ошибка при загрузке (" + e.getMessage() + ")");

}

}

//ОЧИСТКА ОСНОВНОЙ ТАБЛИЦЫ

private void DeleteMain\_Button1ActionPerformed(java.awt.event.ActionEvent evt) {

DefaultTableModel myTable=(DefaultTableModel)CurTable.getModel();

myTable.setRowCount(0);

}

////////////////////////////////////////////////////////////////////////////////////////////////////

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(JFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

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

} catch (IllegalAccessException ex) {

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

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

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

}\*/

try

{

javax.swing.UIManager.setLookAndFeel(new FlatDarkLaf());

}catch(Exception e)

{

}

//</editor-fold>

Collection<String> a;

/\* Create and display the form \*/

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

public void run() {

new JFrame().setVisible(true);

}

});

}

// Variables declaration - do not modify

private javax.swing.JTabbedPane ActionMenu;

private javax.swing.JButton Add\_button;

private javax.swing.JScrollPane CalTable;

private javax.swing.JButton Calcul\_button;

private javax.swing.JButton CleanCont\_Button;

private javax.swing.JTable ConTable;

private javax.swing.JScrollPane ConteinerTable;

private javax.swing.JTable CurTable;

private javax.swing.JScrollPane CurrentTable;

private javax.swing.JLabel DBtext;

private javax.swing.JButton DeleteMain\_Button;

private javax.swing.JButton DeleteMain\_Button1;

private javax.swing.JButton DeteleCont\_button;

private javax.swing.JLabel Logo;

private javax.swing.JTabbedPane MainTable;

private javax.swing.JButton Return\_Button;

private javax.swing.JButton SaveConteiner\_Button;

private javax.swing.JScrollPane SaveTable;

private javax.swing.JLabel Stext;

private javax.swing.JLabel UBtext;

private javax.swing.JFileChooser jFileChooser1;

private javax.swing.JMenu jMenu1;

private javax.swing.JMenu jMenu2;

private javax.swing.JMenu jMenu3;

private javax.swing.JMenuBar jMenuBar1;

private javax.swing.JMenuItem jMenuItem1;

private javax.swing.JMenuItem jMenuItem2;

private javax.swing.JMenuItem jMenuItem3;

private javax.swing.JMenuItem jMenuItem4;

private javax.swing.JPanel jPanel1;

private javax.swing.JPanel jPanel2;

private javax.swing.JTextField lower\_bound\_ent;

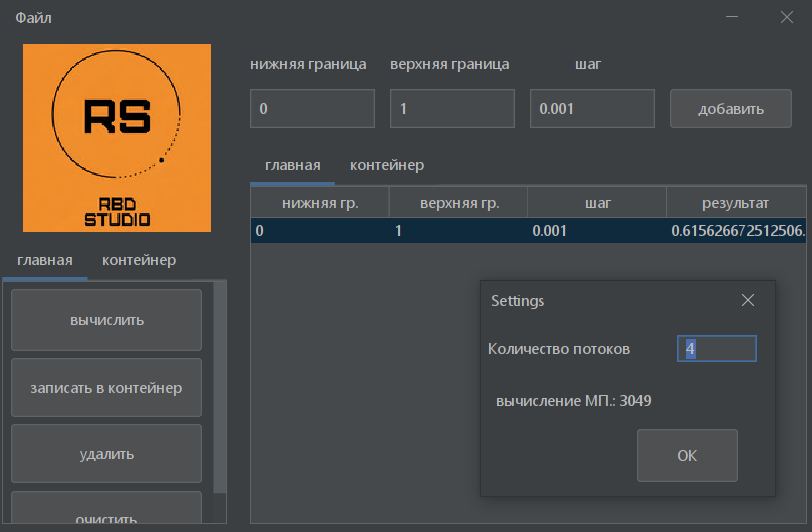
private javax.swing.JTextField step\_ent;

private javax.swing.JTextField upper\_bound\_ent;

// End of variables declaration

}

**Результаты работы программы:**

****

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