Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

ОТЧЁТ

по лабораторной работе №5

по курсу «Программирование на языке Java»

на тему «Многопоточность в Java»

Выполнили:

Студенты группы 22ВВП1

Матросов А.М.

Пяткин М.Э.

Приняли:

Юрова О. В.

Карамышева Н. С.

Пенза 2025

### Цель работы

Научиться создавать многопоточные приложения c использованием стандартных средств языка Java.

### Лабораторное задание:

**Вариант 10**:

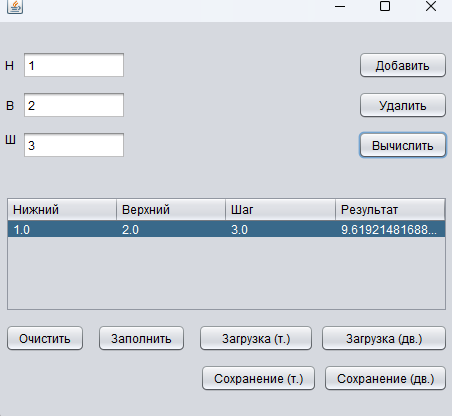
Модифицировать приложение из предыдущей лабораторной работы, реализовав вычисление определенного интеграла в нескольких дополнительных потоках (число потоков определяется номером варианта), снимая нагрузку с основного потока и предотвращая "подвисание" графического интерфейса. Варианты с номерами до 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**

1. Создали класс IntegralCalculator для работы с многопоточностью (уже внутри него происходит вычисление через trapezoidalIntegral). Конструктор класса принимает объект RecIntegral, таблицу и номер строки, и они сохраняются в полях класса для использования в run().  
   Метод run() выполняется при запуске потока. Считаем интеграл, записываем результат в RecIntegral и обновляем GUI
2. Добавили вызов IntegralCalculator при создании нового потока для кнопки «вычислить».

**Результат работы программы**

****

***Рисунок 1 – вычисление интеграла.***

**Вывод:** в ходе выполнения лабораторной научились работать с многопоточностью в языке Java, реализовывать Runnable интерфейс.

**Листинг**

/\*

\* 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

\*/

package my.contacteditor;

import javax.swing.JFileChooser;

import java.io.\*;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

import java.util.LinkedList;

import javax.swing.SwingUtilities;

class InvalidIntegralValueException extends Exception {

public InvalidIntegralValueException(String message) {

super(message);

}

}

class RecIntegral implements Serializable {

private static final long serialVersionUID = 1L;

private double a, b, step, result;

public RecIntegral(double a, double b, double step, double result) throws InvalidIntegralValueException {

if (!isValid(a) || !isValid(b) || !isValid(step)) {

throw new InvalidIntegralValueException("Приемлемый диапазон от 0,000001 до 1000000.");

}

this.a = a;

this.b = b;

this.step = step;

this.result = result;

}

private boolean isValid(double value) {

return value >= 0.000001 && value <= 1000000;

}

// Геттеры

public double getA() {

return a;

}

public double getB() {

return b;

}

public double getStep() {

return step;

}

public double getResult() {

return result;

}

// Сеттер для результата (остальные можно оставить без сеттеров, если значения не должны меняться)

public void setResult(double result) {

this.result = result;

}

}

/\*\*

\*

\* @author AlexMat

\*/

public class ContactEditorUI extends javax.swing.JFrame {

private LinkedList<RecIntegral> records = new LinkedList<>();

/\*\*

\* Creates new form ContactEditorUI

\*/

public ContactEditorUI() {

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() {

jFileChooser1 = new javax.swing.JFileChooser();

jButton1 = new javax.swing.JButton();

jButton2 = new javax.swing.JButton();

jButton3 = new javax.swing.JButton();

jTextField1 = new javax.swing.JTextField();

jTextField2 = new javax.swing.JTextField();

jTextField3 = new javax.swing.JTextField();

jLabel1 = new javax.swing.JLabel();

jLabel2 = new javax.swing.JLabel();

jLabel3 = new javax.swing.JLabel();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

jButton4 = new javax.swing.JButton();

jButton5 = new javax.swing.JButton();

jButton6 = new javax.swing.JButton();

jButton7 = new javax.swing.JButton();

jButton8 = new javax.swing.JButton();

jButton9 = new javax.swing.JButton();

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

jButton1.setText("Добавить");

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

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

jButton1ActionPerformed(evt);

}

});

jButton2.setText("Удалить");

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

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

jButton2ActionPerformed(evt);

}

});

jButton3.setText("Вычислить");

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

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

jButton3ActionPerformed(evt);

}

});

jTextField1.addActionListener(new java.awt.event.ActionListener() {

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

jTextField1ActionPerformed(evt);

}

});

jTextField2.addActionListener(new java.awt.event.ActionListener() {

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

jTextField2ActionPerformed(evt);

}

});

jTextField3.addActionListener(new java.awt.event.ActionListener() {

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

jTextField3ActionPerformed(evt);

}

});

jLabel1.setText("Н");

jLabel2.setText("В");

jLabel3.setText("Ш");

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

new Object [][] {

},

new String [] {

"Нижний", "Верхний", "Шаг", "Результат"

}

));

jScrollPane1.setViewportView(jTable1);

jButton4.setText("Очистить");

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

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

jButton4ActionPerformed(evt);

}

});

jButton5.setText("Заполнить");

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

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

jButton5ActionPerformed(evt);

}

});

jButton6.setText("Загрузка (т.)");

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

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

jButton6ActionPerformed(evt);

}

});

jButton7.setText("Загрузка (дв.)");

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

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

jButton7ActionPerformed(evt);

}

});

jButton8.setText("Сохранение (т.)");

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

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

jButton8ActionPerformed(evt);

}

});

jButton9.setText("Сохранение (дв.)");

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

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

.addGroup(layout.createSequentialGroup()

.addContainerGap()

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 0, Short.MAX\_VALUE))

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

.addContainerGap()

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

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

.addComponent(jButton4)

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

.addComponent(jButton5)

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

.addComponent(jButton6, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

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

.addGroup(layout.createSequentialGroup()

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

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

.addComponent(jLabel1)

.addComponent(jLabel2))

.addComponent(jLabel3))

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

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

.addComponent(jTextField1, javax.swing.GroupLayout.DEFAULT\_SIZE, 104, Short.MAX\_VALUE)

.addComponent(jTextField2)

.addComponent(jTextField3))

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

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

.addComponent(jButton3, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton2, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jButton1, javax.swing.GroupLayout.Alignment.TRAILING, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)))))

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

.addGap(201, 201, 201)

.addComponent(jButton8)

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

.addComponent(jButton9)))

.addContainerGap())

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup()

.addGap(29, 29, 29)

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

.addComponent(jButton1)

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

.addComponent(jLabel1))

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

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

.addComponent(jButton2)

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

.addComponent(jLabel2))

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

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

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

.addComponent(jButton3)

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

.addComponent(jLabel3))

.addGap(37, 37, 37)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 116, javax.swing.GroupLayout.PREFERRED\_SIZE)

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

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

.addComponent(jButton4)

.addComponent(jButton5)

.addComponent(jButton6)

.addComponent(jButton7))

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

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

.addComponent(jButton8)

.addComponent(jButton9))

.addContainerGap(26, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

private double trapezoidalIntegral(double a, double b, double dx) {

int n = (int) Math.ceil((b - a) / dx); // Вычисляем количество разбиений по переданному шагу

double s1 = 0, s2;

do {

s2 = 0;

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

double x = a + i \* dx;

s2 += function(x);

}

s2 = (s2 + function(a) / 2 + function(b) / 2) \* dx;

double temp = s1;

s1 = s2;

s2 = temp;

// Теперь `dx` не меняем!

} while (Math.abs(s1 - s2) > 1e-2); // Возможно, здесь ты имел в виду какую-то точность?

return s1;

}

private double function(double x) {

return Math.exp(x) / x;

}

class IntegralCalculator implements Runnable {

private RecIntegral rec;

private DefaultTableModel model;

private int rowIndex;

public IntegralCalculator(RecIntegral rec, DefaultTableModel model, int rowIndex) {

this.rec = rec;

this.model = model;

this.rowIndex = rowIndex;

}

@Override

public void run() {

double result = trapezoidalIntegral(rec.getA(), rec.getB(), rec.getStep());

rec.setResult(result);

SwingUtilities.invokeLater(() -> model.setValueAt(result, rowIndex, 3));

}

}

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

try {

double a = Double.parseDouble(jTextField1.getText());

double b = Double.parseDouble(jTextField2.getText());

double step = Double.parseDouble(jTextField3.getText());

if (a >= b) {

JOptionPane.showMessageDialog(this, "Некорректные границы интегрирования!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

return;

}

for (RecIntegral rec : records) {

if (rec.getA() == a && rec.getB() == b && rec.getStep() == step) {

JOptionPane.showMessageDialog(this, "Такие данные уже добавлены!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

return;

}

}

RecIntegral newRec = new RecIntegral(a, b, step, -1); // Может выбросить исключение

records.add(newRec);

((DefaultTableModel) jTable1.getModel()).addRow(new Object[]{a, b, step, "-"});

} catch (NumberFormatException e) {

JOptionPane.showMessageDialog(this, "Ошибка ввода! Введите числа.", "Ошибка", JOptionPane.ERROR\_MESSAGE);

} catch (InvalidIntegralValueException e) {

JOptionPane.showMessageDialog(this, e.getMessage(), "Ошибка", JOptionPane.ERROR\_MESSAGE);

}

}

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

int selectedRow = jTable1.getSelectedRow();

if (selectedRow == -1) {

JOptionPane.showMessageDialog(this, "Выберите строку для удаления!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

return;

}

records.remove(selectedRow);

((DefaultTableModel) jTable1.getModel()).removeRow(selectedRow);

}

private void jTextField1ActionPerformed(java.awt.event.ActionEvent evt) {

}

private void jTextField2ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

private void jTextField3ActionPerformed(java.awt.event.ActionEvent evt) {

// TODO add your handling code here:

}

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

int selectedRow = jTable1.getSelectedRow();

if (selectedRow == -1) {

JOptionPane.showMessageDialog(this, "Выберите строку для вычисления!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

return;

}

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

if (!"-".equals(model.getValueAt(selectedRow, 3).toString())) {

JOptionPane.showMessageDialog(this, "Эта строка уже вычислена!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

return;

}

RecIntegral rec = records.get(selectedRow);

// Запускаем вычисление в отдельном потоке

new Thread(new IntegralCalculator(rec, model, selectedRow)).start();

}

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

((DefaultTableModel) jTable1.getModel()).setRowCount(0);

}

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

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

model.setRowCount(0);

for (RecIntegral rec : records) {

model.addRow(new Object[]{rec.getA(), rec.getB(), rec.getStep(), rec.getResult() == -1 ? "-" : rec.getResult()});

}

}

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

// Загрузка из текстового файла

JFileChooser fileChooser = new JFileChooser();

int returnValue = fileChooser.showOpenDialog(this);

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

File file = fileChooser.getSelectedFile();

try (BufferedReader reader = new BufferedReader(new FileReader(file))) {

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

model.setRowCount(0);

records.clear();

String line;

while ((line = reader.readLine()) != null) {

String[] parts = line.split(" ");

if (parts.length == 4) {

double a = Double.parseDouble(parts[0]);

double b = Double.parseDouble(parts[1]);

double step = Double.parseDouble(parts[2]);

double result = Double.parseDouble(parts[3]);

RecIntegral rec = new RecIntegral(a, b, step, result);

records.add(rec);

model.addRow(new Object[]{a, b, step, result == -1 ? "-" : result});

}

}

} catch (IOException | NumberFormatException | InvalidIntegralValueException e) {

JOptionPane.showMessageDialog(this, "Ошибка при загрузке файла!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

}

}

}

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

// Загрузка из двоичного файла

JFileChooser fileChooser = new JFileChooser();

int returnValue = fileChooser.showOpenDialog(this);

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

File file = fileChooser.getSelectedFile();

try (ObjectInputStream ois = new ObjectInputStream(new FileInputStream(file))) {

records = (LinkedList<RecIntegral>) ois.readObject();

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

model.setRowCount(0);

for (RecIntegral rec : records) {

model.addRow(new Object[]{rec.getA(), rec.getB(), rec.getStep(), rec.getResult() == -1 ? "-" : rec.getResult()});

}

} catch (IOException | ClassNotFoundException e) {

JOptionPane.showMessageDialog(this, "Ошибка при загрузке бинарного файла!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

}

}

}

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

// Сохранение в текстовый файл

JFileChooser fileChooser = new JFileChooser();

int returnValue = fileChooser.showSaveDialog(this);

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

File file = fileChooser.getSelectedFile();

try (BufferedWriter writer = new BufferedWriter(new FileWriter(file))) {

for (RecIntegral rec : records) {

writer.write(rec.getA() + " " + rec.getB() + " " + rec.getStep() + " " + rec.getResult());

writer.newLine();

}

} catch (IOException e) {

JOptionPane.showMessageDialog(this, "Ошибка при сохранении файла!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

}

}

}

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

// Сохранение в двоичный файл

JFileChooser fileChooser = new JFileChooser();

int returnValue = fileChooser.showSaveDialog(this);

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

File file = fileChooser.getSelectedFile();

try (ObjectOutputStream oos = new ObjectOutputStream(new FileOutputStream(file))) {

oos.writeObject(records);

} catch (IOException e) {

JOptionPane.showMessageDialog(this, "Ошибка при сохранении бинарного файла!", "Ошибка", JOptionPane.ERROR\_MESSAGE);

}

}

}

/\*\*

\* @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(ContactEditorUI.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);

} catch (InstantiationException ex) {

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

} catch (IllegalAccessException ex) {

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

} catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(ContactEditorUI.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 ContactEditorUI().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.JFileChooser jFileChooser1;

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

}