Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

Пенза 2021

**ОТЧЁТ**

по лабораторной работе №1

по курсу «Программирование на языке JAVA»

на тему «Графические интерфейсы»

Выполнили:

студенты группы 21ВВП1

Брянцев А. Г.

Сущев М. В.

Тюрин В. И.

Приняли:

Юрова О.В.

Карамышева Н.С.

Пенза 2024

**Цель работы:** научиться разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing.

Вариант – 9 ( cos (x^2) )

**Результат работы программы:**

**Изображение выглядит как текст, снимок экрана, программное обеспечение, Значок на компьютере

Автоматически созданное описание**

**Изображение выглядит как текст, снимок экрана, Шрифт, веб-страница

Автоматически созданное описание**

**Листинг**

import javax.swing.\*;  
import javax.swing.table.DefaultTableModel;  
import java.awt.event.ActionEvent;  
import java.awt.event.ActionListener;  
  
public class IntegralCalculatorGUI extends JFrame {  
  
 private DefaultTableModel tableModel;  
 private JTable table;  
 private JTextField lowerLimitField;  
 private JTextField upperLimitField;  
 private JTextField stepField;  
  
 public IntegralCalculatorGUI() {  
 setTitle("Integral Calculator");  
 setSize(600, 400);  
 setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);  
  
 tableModel = new DefaultTableModel();  
 tableModel.addColumn("Lower Limit");  
 tableModel.addColumn("Upper Limit");  
 tableModel.addColumn("Step");  
 tableModel.addColumn("Result");  
// создание таблицы  
 table = new JTable(tableModel);  
  
 lowerLimitField = new JTextField(2);  
 upperLimitField = new JTextField(2);  
 stepField = new JTextField(2);  
  
 JButton addButton = new JButton("Add");  
 addButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 tableModel.addRow(new Object[]{lowerLimitField.getText(), upperLimitField.getText(), stepField.getText(), ""});  
 }  
 });  
  
 JButton deleteButton = new JButton("Delete");  
 deleteButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 int selectedRow = table.getSelectedRow();  
 if (selectedRow != -1) {  
 tableModel.removeRow(selectedRow);  
 }  
 }  
 });  
  
 JButton calculateButton = new JButton("Calculate");  
 //проверка введеных значений  
 calculateButton.addActionListener(new ActionListener() {  
 @Override  
 public void actionPerformed(ActionEvent e) {  
 double lowerLimit, upperLimit, step;  
  
 try {  
 lowerLimit = Double.parseDouble(lowerLimitField.getText());  
 upperLimit = Double.parseDouble(upperLimitField.getText());  
 step = Double.parseDouble(stepField.getText());  
  
 double integralResult = calculateIntegral(lowerLimit, upperLimit, step);  
  
 int selectedRow = table.getSelectedRow();  
 if (selectedRow != -1) {  
 tableModel.setValueAt(integralResult, selectedRow, 3);  
 }  
 } catch (NumberFormatException ex) {  
 JOptionPane.showMessageDialog(null, "Please enter valid numerical values.");  
 }  
 }  
 });  
// создание полей и кнопок  
 JPanel inputPanel = new JPanel();  
 inputPanel.add(new JLabel("Lower Limit:"));  
 inputPanel.add(lowerLimitField);  
 inputPanel.add(new JLabel("Upper Limit:"));  
 inputPanel.add(upperLimitField);  
 inputPanel.add(new JLabel("Step:"));  
 inputPanel.add(stepField);  
 inputPanel.add(addButton);  
 inputPanel.add(deleteButton);  
 inputPanel.add(calculateButton);  
  
 JPanel mainPanel = new JPanel();  
 mainPanel.add(inputPanel);  
 mainPanel.add(new JScrollPane(table));  
  
 getContentPane().add(mainPanel);  
 }  
  
 // вычисление интеграла cos(x^2)  
 public static double calculateIntegral(double lowerLimit, double upperLimit, double step) {  
 double x1, x2, sum = 0;  
 int amountSteps = (int)((upperLimit - lowerLimit) / step); //округляется в меньшую сторону  
 x1 = lowerLimit;  
  
 for (int i = 0; i < amountSteps; i++){  
 x2 = x1 + step;  
 sum += 0.5 \* step \* (Math.cos(x1\*x1) + Math.cos(x2\*x2));  
 x1 = x2;  
 }  
 if ((upperLimit - lowerLimit) % step != 0)  
 sum += 0.5 \* (upperLimit - x1) \* (Math.cos(x1\*x1) + Math.cos(upperLimit\*upperLimit));  
  
 return sum;  
 }  
  
  
 // запуск GUI  
 public static void main(String[] args) {  
 SwingUtilities.invokeLater(() -> {  
 IntegralCalculatorGUI gui = new IntegralCalculatorGUI();  
 gui.setVisible(true);  
 });  
 }  
}

**Вывод:** в ходе данной лабораторной работы научились разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing. Были получены знания по программированию сложных математических функций и расчету их площади методом трапеций. Было выявлено, что с уменьшением шага вычислений, увеличивается точность расчета.