Пензенский государственный университет

Факультет вычислительной техники

Кафедра «Вычислительная техника»

Отчет о лабораторной работе №1

по дисциплине «Разработка кроссплатформенных приложений»

на тему «Графические интерфейсы»

Выполнили: ст-ты гр. 22ВОЭ1

Старёхин Б.А.

Николаев Д.А.

Проверили: к.т.н., доцент каф. ВТ

Юрова О.В.

Карамышева Н.С.

2025

Цель работы

Научиться разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing.

Задание

Вычислить определенный интеграл функции e-x. Разработать приложение, обладающее графическим интерфейсом с использованием языка Java и библиотеки Swing. Приложение должно содержать 3 поля ввода (JTextField), доступных для редактирования, и соответственно таблицу (JTable) с четырьмя колонками: нижняя граница интегрирования, верхняя граница интегрирования, шаг интегрирования и результат вычисления. Кроме того, должны присутствовать 3 кнопки (JButton): добавить, удалить, вычислить. Для добавления/удаления строки и вычисления значения определенного интеграла для функции e-x и параметров выделенной строки таблицы. Результат должен выводиться в четвертой колонке, которая не доступна для редактирования.

Ход работы

Листинг программы

package my.contacteditor;

import javax.swing.table.DefaultTableModel;

import javax.swing.JOptionPane;

public class ContactEditorUI extends javax.swing.JFrame {

public ContactEditorUI() {

initComponents();

}

@SuppressWarnings("unchecked")

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

private void initComponents() {

jPanel1 = new javax.swing.JPanel();

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();

jScrollPane1 = new javax.swing.JScrollPane();

jTable1 = new javax.swing.JTable();

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

jLabel1.setText("верхний предел");

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

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

jTextField1ActionPerformed(evt);

}

});

jLabel2.setText("нижний предел");

jLabel3.setText("шаг");

jButton1.setText("добавить");

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

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

jButton1ActionPerformed(evt);

}

});

jButton2.setText("вычислить");

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

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

jButton2ActionPerformed(evt);

}

});

jButton3.setText("удалить");

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

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

jButton3ActionPerformed(evt);

}

});

jTable1.setFont(new java.awt.Font("Tahoma", 1, 12)); // NOI18N

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

new Object [][] {

},

new String [] {

"верх.пред", "ниж.пред", "шаг", "результат"

}

) {

boolean[] canEdit = new boolean [] {

false, false, false, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

jScrollPane1.setViewportView(jTable1);

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, false)

.addComponent(jLabel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

.addComponent(jTextField1)

.addComponent(jLabel2)

.addComponent(jTextField2)

.addComponent(jLabel3)

.addComponent(jTextField3))

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

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

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

.addComponent(jButton3, 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))

.addGap(92, 92, 92))

.addGroup(javax.swing.GroupLayout.Alignment.TRAILING, jPanel1Layout.createSequentialGroup()

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

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 375, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addContainerGap())

);

jPanel1Layout.setVerticalGroup(

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

.addGroup(jPanel1Layout.createSequentialGroup()

.addContainerGap()

.addComponent(jLabel1)

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

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

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

.addComponent(jButton1))

.addGap(18, 18, 18)

.addComponent(jLabel2)

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

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

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

.addComponent(jButton2))

.addGap(18, 18, 18)

.addComponent(jLabel3)

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

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

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

.addComponent(jButton3))

.addGap(18, 18, 18)

.addComponent(jScrollPane1, javax.swing.GroupLayout.DEFAULT\_SIZE, 126, Short.MAX\_VALUE)

.addContainerGap())

);

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

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

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

.addComponent(jPanel1, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, Short.MAX\_VALUE)

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup()

.addComponent(jPanel1, javax.swing.GroupLayout.PREFERRED\_SIZE, javax.swing.GroupLayout.DEFAULT\_SIZE, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(0, 0, Short.MAX\_VALUE))

);

pack();

}// </editor-fold>

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

}

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

double vpred;

double npred;

double step;

vpred = Double.parseDouble(jTextField1.getText());

npred = Double.parseDouble(jTextField2.getText());

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

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

tModel.addRow(new Object[] {vpred, npred, step});

jTextField1.setText("");

jTextField2.setText("");

jTextField3.setText("");

}

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

double vp;

double np;

double st;

double result;

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

int rowNom = jTable1.getSelectedRow();

if (rowNom == -1) {

JOptionPane.showMessageDialog(null, "выберите строку!");

}

else {

vp = Double.parseDouble(tModel.getValueAt(rowNom, 0).toString());

np = Double.parseDouble(tModel.getValueAt(rowNom, 1).toString());

st = Double.parseDouble(tModel.getValueAt(rowNom, 2).toString());

result = CalcInt (vp,np,st);

tModel.setValueAt(result, rowNom, 3);

}

}

public double CalcInt (double vp, double npr, double step){

double start, h, sumS=0;

if (vp > npr) {

double temp = vp;

vp = npr;

npr = temp;

}

start = vp;

while (start > npr) {

h = Math.min(step, start - npr);

sumS += h \* (Math.exp(-start) + Math.exp(-(start - h))) / 2;

start -= h;

}

return sumS;

}

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

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

int rowNom = jTable1.getSelectedRow();

if (rowNom == -1) {

JOptionPane.showMessageDialog(null, "выберите строку!");

}

else {

tModel.removeRow(rowNom);

}

}

public static void main(String args[]) {

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.JLabel jLabel1;

private javax.swing.JLabel jLabel2;

private javax.swing.JLabel jLabel3;

private javax.swing.JPanel jPanel1;

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

}

Словесное описание алгоритма

1. Объявление метода, который обрабатывает событие нажатия кнопки jButton1.
2. Объявление трех переменных типа double: vpred, npred и step, которые будут использоваться для хранения значений.
3. Получение текста из текстового поля jTextField1, преобразование его в число типа double и присвоение переменной vpred.
4. Получение текста из jTextField2, преобразование в double и присвоение переменной npred.
5. Получение текста из jTextField3, преобразование в double и присвоение переменной step.
6. Получение модели таблицы из jTable1 и приведение ее к типу DefaultTableModel, чтобы можно было добавлять строки.
7. Добавление новой строки в модель таблицы с тремя значениями: vpred, npred и step.
8. Очистка текстовых полей после добавления данных в таблицу.
9. Объявление метода, который обрабатывает событие нажатия кнопки jButton2.
10. Объявление четырех переменных типа double: vp, np, st и result, которые будут использоваться для хранения значений из выбранной строки таблицы и результата вычислений.
11. Получение модели таблицы из jTable1 и приведение ее к типу DefaultTableModel.
12. Получение номера выбранной строки в таблице. Если строка не выбрана, rowNom будет равен -1.
13. Проверка, выбрана ли строка. Если нет, отображается сообщение с просьбой выбрать строку.
14. Если строка выбрана, извлечение значений из выбранной строки таблицы и преобразование их в double. Значения присваиваются переменным vp, np и st.
15. Вызов метода CalcInt, который выполняет вычисления с использованием значений vp, np и st, и присвоение результата переменной result.
16. Установка значения result в четвертую колонку (индекс 3) выбранной строки таблицы.
17. Объявление метода CalcInt, который принимает три параметра: vp, npr и step. Метод возвращает значение типа double.
18. Объявление переменных:

start — начальное значение, инициализируется значением vp.

h — переменная для хранения текущего шага.

sumS — переменная для накопления суммы, инициализируется нулем.

1. Инициализация переменной start значением vp.
2. Начало цикла do-while, который будет выполняться хотя бы один раз.
3. Вычисление значения h, которое будет равно меньшему из step и разности npr и start. Это гарантирует, что шаг не превысит оставшееся расстояние до npr.
4. Обновление sumS с использованием метода трапеций для численного интегрирования. Здесь вычисляется площадь трапеции, основанной на текущем start и start + h.
5. Увеличение start на значение h, чтобы перейти к следующему интервалу.
6. Условие продолжения цикла: пока start меньше npr, цикл будет выполняться.
7. Возврат накопленной суммы sumS, которая представляет собой результат интегрирования.
8. Объявление метода, который обрабатывает событие нажатия кнопки jButton3.
9. Получение модели таблицы jTable1 и приведение её к типу DefaultTableModel, чтобы можно было изменять данные в таблице.
10. Получение номера выбранной строки в таблице. Если строка не выбрана, rowNom будет равен -1.
11. Проверка, выбрана ли строка.
12. Показ сообщения пользователю с просьбой выбрать строку.
13. Удаление выбранной строки из модели таблицы.

Результат выполнения программы

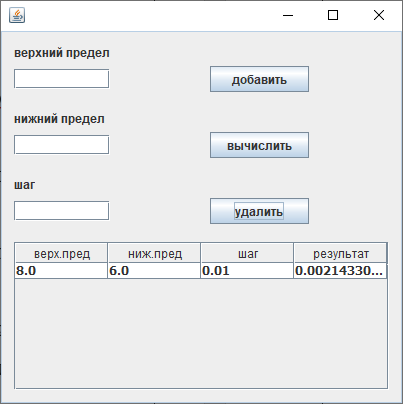


Рисунок 1 — Добавление значений в таблицу

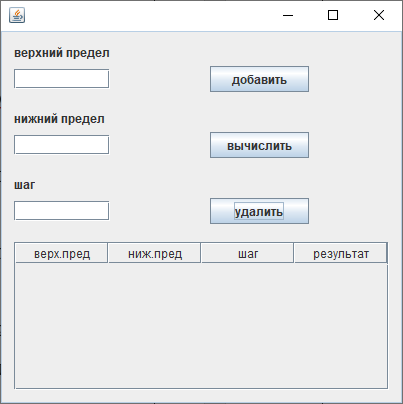


Рисунок 2 — Значения из таблицы были удалены

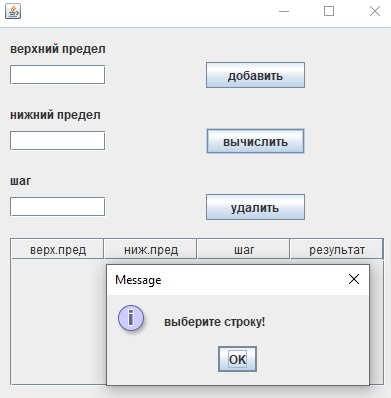


Рисунок 3 — Всплывающее окно, если не была выбрана строчка

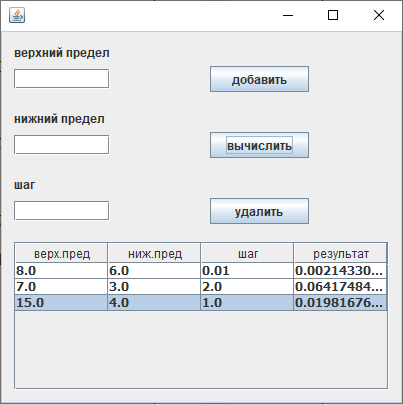


Рисунок 4 — Вычисление результата

Ручной просчет

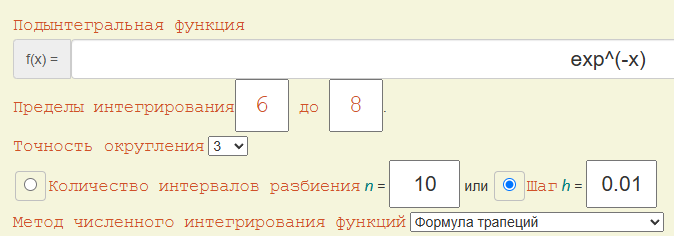




Рисунок 5 — Первый просчет

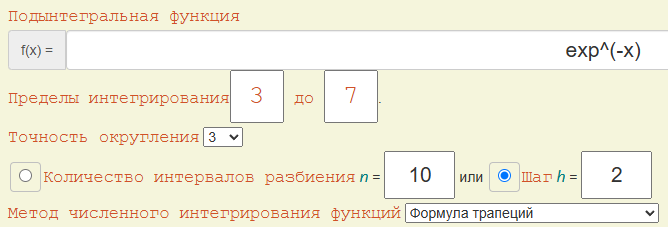




Рисунок 6 — Второй просчет

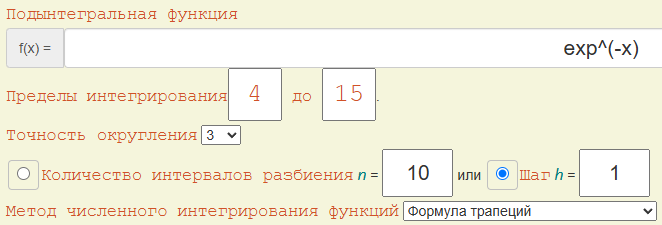




Рисунок 7 — Третий просчет

Результаты выполнения программы совпадают с ручным просчетом, следовательно, программа работает правильно.

Вывод

В данной лабораторной работе были получены навыки для того, чтобы разрабатывать приложения, обладающие графическим интерфейсом пользователя, с использованием библиотеки Swing.