Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе № 1

по дисциплине: «Программирование на языке Java»

на тему: «Графические интерфейсы»

Выполнили:

студенты группы 20ВВП1

Макарова А. Ю.

Тельнова А. Д.

Приняли:

к.т.н., доцент

Юрова О. В.

к.т.н., доцент  
Карамышева Н.С.

Пенза, 2023

# Цель работы

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

# Лабораторное задание

Необходимо вычислить определенный интеграл функции по формуле f = sin(x).

Разработать приложение, обладающее графическим интерфейсом с использованием языка Java и библиотеки Swing. Приложение должно содержать три поля ввода (JTextField), доступных для редактирования, и таблицу (JTable) с четырьмя колонками: нижняя граница интегрирования, верхняя граница интегрирования, шаг интегрирования и результат вычисления. Должны присутствовать три кнопки (JButton): добавить, удалить, вычислить. Результат должен выводиться в четвертой колонке, которая не доступна для редактирования.

# Описание работы программы

Создан контейнер JFrame, в него добавлены элементы интерфейса, такие как JButton, JTable, JTextField, JLabel.

На форме размещены три текстовых поля, которые имеют подписи: «Нижняя граница», «Верхняя граница», «Шаг». Под текстовыми полями располагаются кнопки: «Добавить», «Удалить», «Вычислить». В самом низу формы располагается таблица с четырьмя колонками: «Нижняя граница», «Верхняя граница», «Шаг», «Результат».

Кнопка «Добавить» добавляет в таблицу введенные данные из текстового поля. Кнопка «Удалить» удаляет выбранную строку из таблицы. Кнопка «Вычислить» вычисляет значение определенного интеграла и заносит результат в четвертый столбец таблицы. Первые три столбца таблицы доступны для редактирования, четвертый столбец недоступен для редактирования.

Для каждой кнопки добавлен обработчик события.

Метод private void AddButtonActionPerformed обрабатывает нажатие на кнопку «Добавить» (AddButton). Если не заполнено хотя бы одно из полей «Нижняя граница», «Верхняя граница» или «Шаг», то будет выведено сообщение об ошибке с просьбой заполнить все текстовые поля. Если все поля заполнены, но данные некорректны (например, шаг равен нулю или нижняя граница интегрирования больше верхней), то так же будет выведено сообщение о том, что данные введены неверно. Если данные введены корректно, нажатие на кнопку добавляет данные из текстовых полей в соответствующие столбцы таблицы.

Метод private void DeleteButtonActionPerformed обрабатывает нажатие на кнопку «Удалить» (DeleteButton). При нажатии на кнопку из таблицы удаляется выбранная строка. Если ни одна строка не выбрана, то будет выведено сообщение об ошибке с просьбой выбрать желаемую строку для удаления.

Метод private void CalcButtonActionPerformed обрабатывает нажатие на кнопку «Вычислить» (CalcButton). Если ни одна строка не выбрана, то будет выведено сообщение об ошибке с просьбой выбрать желаемую строку для вычисления. В случае, когда данные в таблице будут изменены, но при этом некорректно, то будет выведено сообщение об ошибке. Иначе, в случае, когда была выбрана желаемая строка для вычисления значения определенного интеграла и нажата кнопка «Вычислить», в четвертом столбце появится результат вычисления.

Для вычисления значения определенного интеграла используется метод трапеций. Метод заключается в том, что заданный интервал интегрирования разбивается на n подынтервалов равной длины (длина подынтервала равна шагу). На каждом отрезке дуга графика подынтегральной функции заменяется стягивающей ее хордой и вычисляется площадь трапеций.

Вычисление значения определенного интеграла происходит в цикле до тех пор, пока нижняя граница не превышает верхнюю. Вычисление происходит путем нахождения очередной площади трапеции и суммирования с переменной IntegralFunc, в которой хранится результат вычислений: IntegralFunc += ((Math.sin(i) + Math.sin(i+step))/2)\*step. Если произошел выход за верхнюю границу интегрирования, то площадь последней трапеции вычисляется следующим образом: IntegralFunc += ((Math.sin(up) + Math.sin(i-step))/2)\*step. Далее результат вычисления заносится в последний столбец таблицы.

Были проведены необходимые тесты для проверки правильности работы приложения. Результаты, полученные в программе, были сравнены с ручным подсчетом. Чем меньше шаг, тем точнее результат. Результат вычисления программы совпадает с ручным просчетом.

# Листинг программы

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;

public class JFrame extends javax.swing.JFrame {

public JFrame() {

initComponents();

}

@SuppressWarnings("unchecked")

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

private void initComponents() {

jScrollPane1 = new javax.swing.JScrollPane();

Table = new javax.swing.JTable();

AddButton = new javax.swing.JButton();

DeleteButton = new javax.swing.JButton();

CalcButton = new javax.swing.JButton();

LowTextField = new javax.swing.JTextField();

UpTextField = new javax.swing.JTextField();

StepTextField = new javax.swing.JTextField();

LowLabel = new javax.swing.JLabel();

UpLabel = new javax.swing.JLabel();

StepLabel = new javax.swing.JLabel();

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

jScrollPane1.setBackground(new java.awt.Color(255, 255, 255));

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

new Object [][] {

},

new String [] {

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

}

) {

boolean[] canEdit = new boolean [] {

true, true, true, false

};

public boolean isCellEditable(int rowIndex, int columnIndex) {

return canEdit [columnIndex];

}

});

Table.setSelectionBackground(new java.awt.Color(255, 20, 147));

Table.setShowGrid(false);

Table.getTableHeader().setReorderingAllowed(false);

jScrollPane1.setViewportView(Table);

AddButton.setBackground(new java.awt.Color(255, 20, 147));

AddButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

AddButton.setForeground(new java.awt.Color(255, 255, 255));

AddButton.setText("Добавить");

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

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

AddButtonActionPerformed(evt);

}

});

DeleteButton.setBackground(new java.awt.Color(255, 20, 147));

DeleteButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

DeleteButton.setForeground(new java.awt.Color(255, 255, 255));

DeleteButton.setText("Удалить");

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

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

DeleteButtonActionPerformed(evt);

}

});

CalcButton.setBackground(new java.awt.Color(255, 20, 147));

CalcButton.setFont(new java.awt.Font("Segoe UI", 1, 18)); // NOI18N

CalcButton.setForeground(new java.awt.Color(255, 255, 255));

CalcButton.setText("Вычислить");

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

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

CalcButtonActionPerformed(evt);

}

});

LowTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

UpTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

StepTextField.setFont(new java.awt.Font("Segoe UI", 0, 18)); // NOI18N

LowLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

LowLabel.setForeground(new java.awt.Color(204, 0, 102));

LowLabel.setText("Нижняя граница");

UpLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

UpLabel.setForeground(new java.awt.Color(204, 0, 102));

UpLabel.setText("Верхняя граница");

StepLabel.setFont(new java.awt.Font("sansserif", 0, 24)); // NOI18N

StepLabel.setForeground(new java.awt.Color(204, 0, 102));

StepLabel.setText("Шаг");

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

getContentPane().setLayout(layout);

layout.setHorizontalGroup(

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

.addGroup(layout.createSequentialGroup()

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

.addGroup(layout.createSequentialGroup()

.addGap(26, 26, 26)

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

.addComponent(jScrollPane1)

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

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

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

.addComponent(UpLabel, javax.swing.GroupLayout.Alignment.TRAILING)

.addComponent(StepLabel, javax.swing.GroupLayout.Alignment.TRAILING))

.addGap(18, 18, 18)

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

.addComponent(UpTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 350, Short.MAX\_VALUE)

.addComponent(StepTextField)))

.addGroup(layout.createSequentialGroup()

.addGap(197, 197, 197)

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

.addGroup(layout.createSequentialGroup()

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

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

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

.addGroup(layout.createSequentialGroup()

.addGap(0, 34, Short.MAX\_VALUE)

.addComponent(LowLabel)

.addGap(18, 18, 18)

.addComponent(LowTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 350, javax.swing.GroupLayout.PREFERRED\_SIZE)))

.addGap(24, 24, 24))

);

layout.setVerticalGroup(

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

.addGroup(layout.createSequentialGroup()

.addGap(14, 14, 14)

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

.addComponent(LowTextField, javax.swing.GroupLayout.DEFAULT\_SIZE, 40, Short.MAX\_VALUE)

.addComponent(LowLabel))

.addGap(18, 18, 18)

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

.addComponent(UpTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(UpLabel))

.addGap(18, 18, 18)

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

.addComponent(StepTextField, javax.swing.GroupLayout.PREFERRED\_SIZE, 40, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(StepLabel))

.addGap(33, 33, 33)

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

.addComponent(AddButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(DeleteButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addComponent(CalcButton, javax.swing.GroupLayout.PREFERRED\_SIZE, 50, javax.swing.GroupLayout.PREFERRED\_SIZE))

.addGap(37, 37, 37)

.addComponent(jScrollPane1, javax.swing.GroupLayout.PREFERRED\_SIZE, 230, javax.swing.GroupLayout.PREFERRED\_SIZE)

.addGap(21, 21, 21))

);

pack();

}// </editor-fold>

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

if (LowTextField.getText().equals("") || UpTextField.getText().equals("") || StepTextField.getText().equals("")){

JOptionPane.showMessageDialog(this, "Заполните все поля данными!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

} else

//Double.valueOf(StepTextField.getText()) > Double.valueOf(UpTextField.getText()) ||

if( Double.valueOf(LowTextField.getText())> Double.valueOf(UpTextField.getText()) || Double.valueOf(StepTextField.getText()) == 0)

JOptionPane.showMessageDialog(this, "Что-то не то с данными!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

else

{

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

model.addRow(new Object[]{LowTextField.getText(),UpTextField.getText(),StepTextField.getText()});

LowTextField.setText("");

UpTextField.setText("");

StepTextField.setText("");

}

}

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

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

if (Table.getSelectedRowCount() == 1 )

model.removeRow(Table.getSelectedRow());

else

JOptionPane.showMessageDialog(this, "Выберите строку, которую нужно удалить!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}

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

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

if (Table.getSelectedRowCount() == 1 )

{

double IntegralFunc = 0;

double low = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),0).toString());

double up = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),1).toString());

double step = Double.parseDouble(model.getValueAt(Table.getSelectedRow(),2).toString());

if(low > up || step == 0)

JOptionPane.showMessageDialog(this, "Вы некорректно изменили данные в таблице!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

for (double i = low; i < up - step; i+= step)

{

if (up - low > step)

IntegralFunc += ((Math.sin(i) + Math.sin(i+step))/2)\*step;

else

IntegralFunc += ((Math.sin(up) + Math.sin(i-step))/2)\*step;

}

model.setValueAt(IntegralFunc, Table.getSelectedRow(), 3);

}

else

JOptionPane.showMessageDialog(this, "Выберите строку для вычисления!", "Ошибочка вышла!", JOptionPane.PLAIN\_MESSAGE);

}

public static void main(String args[]) {

//<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);

}

//</editor-fold>

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

public void run() {

JFrame frame = new JFrame();

frame.setVisible(true);

frame.setLocationRelativeTo(null);

frame.setTitle("Вычисляем интегральчик");

frame.getContentPane().setBackground(new java.awt.Color(255, 228, 225));

}

});

}

// Variables declaration - do not modify

private javax.swing.JButton AddButton;

private javax.swing.JButton CalcButton;

private javax.swing.JButton DeleteButton;

private javax.swing.JLabel LowLabel;

private javax.swing.JTextField LowTextField;

private javax.swing.JLabel StepLabel;

private javax.swing.JTextField StepTextField;

private javax.swing.JTable Table;

private javax.swing.JLabel UpLabel;

private javax.swing.JTextField UpTextField;

private javax.swing.JScrollPane jScrollPane1;

// End of variables declaration

}

# Результат работы программы

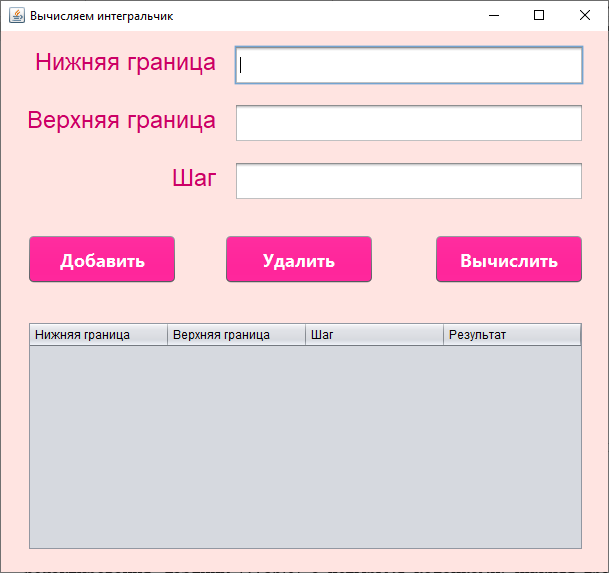


Рисунок 1 – Результат запуска программы

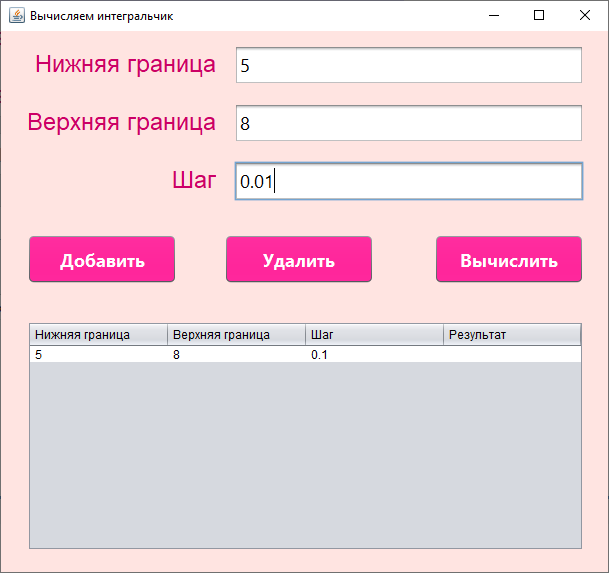


Рисунок – Добавление в таблицу строки с данными

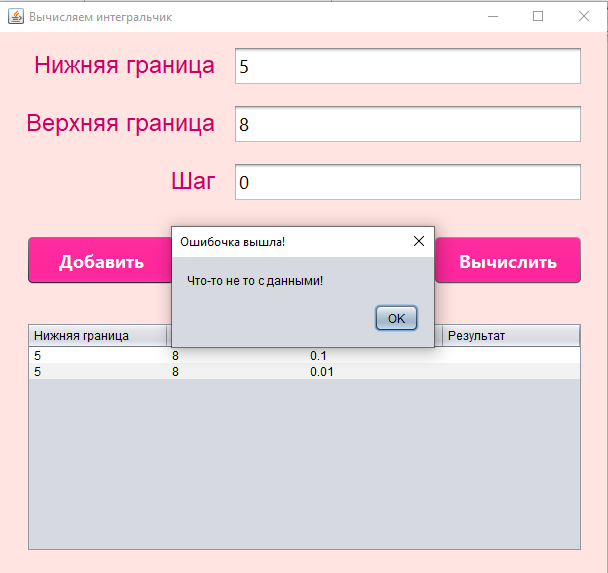


Рисунок – Сообщение об ошибке при вводе некорректных данных

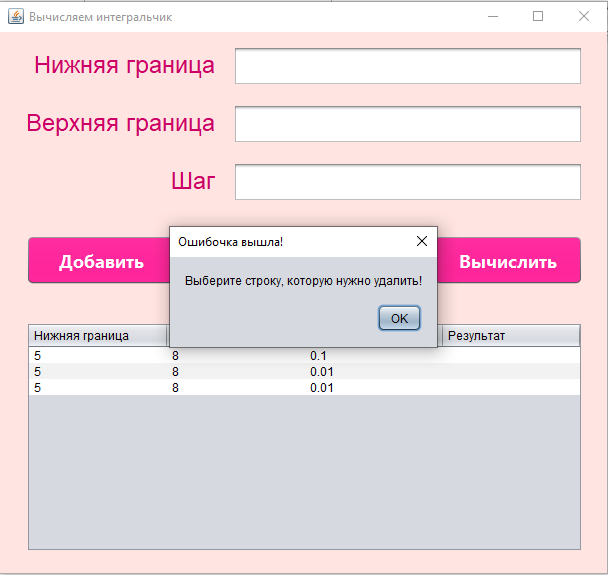


Рисунок – Сообщение об ошибке при попытке удалить строку

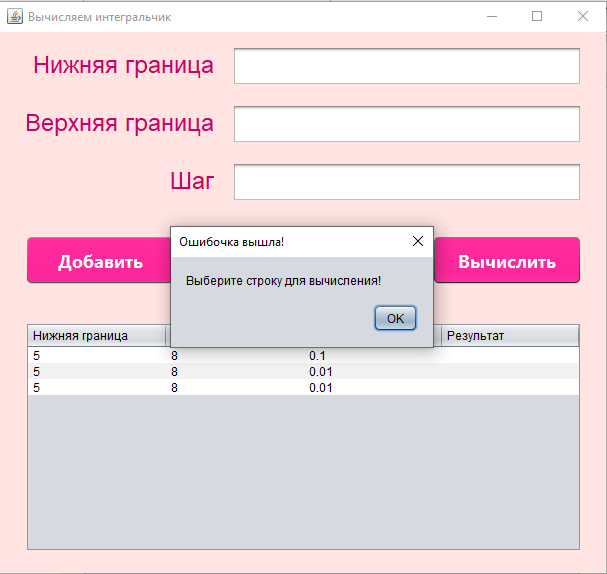


Рисунок – Сообщение об ошибке при попытке вычислить значение функции

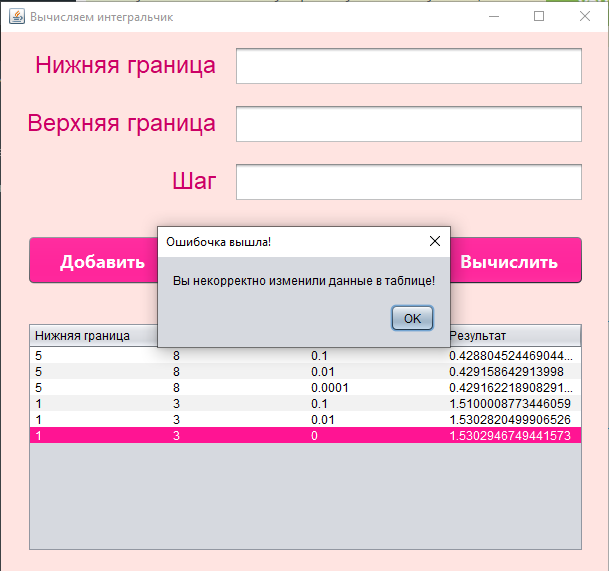


Рисунок – Сообщение об ошибке при некорректном изменении данных в таблице

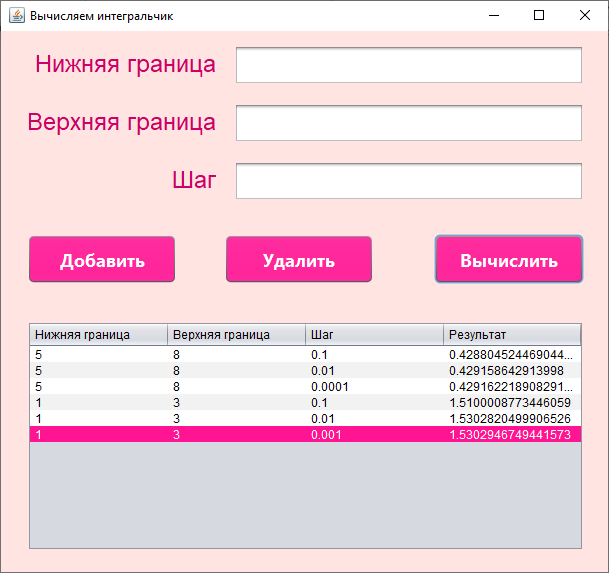


Рисунок – Результаты вычисления функции

# Ручной расчет функции

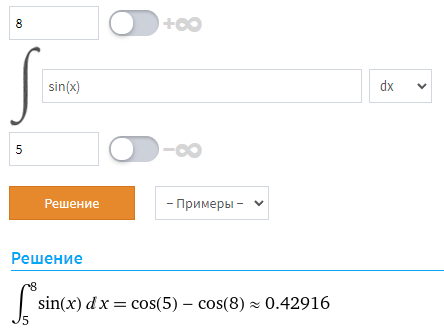


Рисунок – Результат вычисления функции

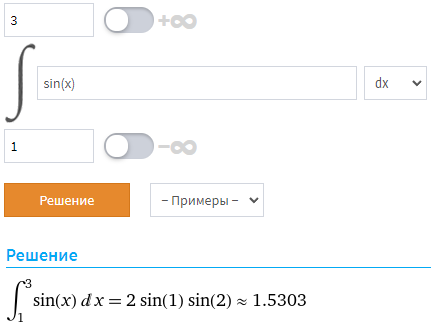


Рисунок – Результат вычисления функции

# Вывод

Разработано приложение, обладающее графическим интерфейсом пользователя, с использованием библиотеки Swing на языке Java для вычисления определенного интеграла функции по формуле f = sin(x).

Приложение содержит поля ввода, доступные для редактирования, таблицу с четырьмя колонками, а также кнопки. Результат вычисления выводится в четвертой колонке таблицы, которая не доступна для редактирования.

Результаты вычислений, полученные в результате работы программы, совпадают с ожидаемыми.