Министерство науки и образования Российской Федерации  
Пензенский государственный университет  
Кафедра “Вычислительная техника”

**ОТЧЁТ**  
по лабораторной работе №1  
по курсу “Программирование на языке Java”  
на тему “Графические интерфейсы”  
Вариант 3

Выполнили: ст. гр. 21ВВП1  
Алёшина А.В.  
Нефёдова Е.Д.  
Сорокина Е.А.

Приняли:  
Юрова О.В.  
Карамышева Н.С.

Пенза 2024

**Цель работы:** изучить библиотеку стандартных коллекций Java Collections Framework, позволяющую хранить различные структуры данных.

**Лабораторное задание:** cos(x)

**Ход работы:**

При нажатии на кнопку “Add” значения из текстовых полей “Lower bound”, “Upper bound” и “Step” добавляются в соответствующие столбцы таблицы. После этого текстовые поля очистятся.

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

// TODO add your handling code here:

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

Table.insertRow(0, new Object[]{jTextFieldDown.getText(), jTextFieldUp.getText(), jTextFieldStep.getText()});

jTextFieldDown.setText("");

jTextFieldUp.setText("");

jTextFieldStep.setText("");

}

При нажатии на кнопку “Delete” из таблицы будет удалена выбранная строка.

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

// TODO add your handling code here:

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

if(jTable1.getSelectedRowCount()==1){

Table.removeRow(jTable1.getSelectedRow());

}

}

При нажатии на кнопку “Calculate” будет рассчитан определенный интеграл выбранной строки таблицы с заданным шагом интегрирования.

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

// TODO add your handling code here:

int selectedRow = jTable1.getSelectedRow();

double down = Double.parseDouble(jTable1.getValueAt(selectedRow, 0).toString());

double up = Double.parseDouble(jTable1.getValueAt(selectedRow, 1).toString());

double step = Double.parseDouble(jTable1.getValueAt(selectedRow, 2).toString());

double calc = 0;

for (double count = down; count < up; count += step){

double count2 = count + step;

if (count2 > up){

count2 = up;

}

double res1 = Math.cos(count);

double res2 = Math.cos(count2);

double res = ((res1 + res2)/2) \* step;

calc += res;

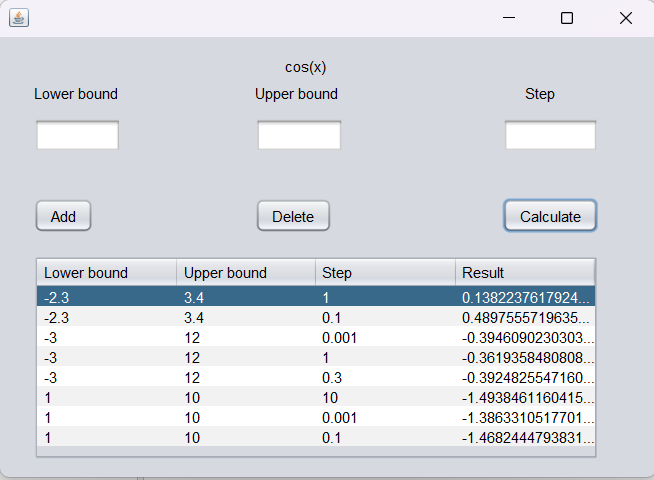
}

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

Table.setValueAt(calc, selectedRow, 3);

}

**Результат выполнения программы для разных случаев:**



**Вывод:**  изучили библиотеку стандартных коллекций Java Collections Framework, позволяющую хранить различные структуры данных.