МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования

**«Вятский государственный университет»**

**(ФГБОУ ВО «ВятГУ»)**

Факультет автоматики и вычислительной техники

Кафедра электронных вычислительных машин

**Разработка программных систем**

Разработка пользовательского интерфейса с использований технологий Swing

Вариант 4

Выполнил студент группы ИВТ-31 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Кудяшев Я.Ю./

Проверил преподаватель\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_/Чистяков Г.А./

Киров 2022

1. Задание

В ходе выполнения лабораторной работы необходимо разработать калькулятор для работы с длинной дробной арифметикой; взаимодействие с пользователем должно осуществляться с применением пользовательского интерфейса.

1. Листинг программы

Листинг программы приведен в приложении А.

1. Экранные формы

Экранные формы приведены в приложении Б.

1. Вывод

В ходе выполнения лабораторной работы были изучены основы технологии Swing, её основные компоненты. Написано приложение с графическим пользовательским интерфейсом, предназначенное для работы с длинной дробной арифметикой.

Приложение А

(обязательное)

Листинг программы

**Main.java**

package com.company;

public class Main {

public static void main(String[] args) {

BigFractionCalculatorGUI app = new BigFractionCalculatorGUI();

app.setVisible(true);

}

}

**BigFractionCalculatorGUI.java**

package com.company;

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

import java.math.BigInteger;

public class BigFractionCalculatorGUI extends JFrame {

/\*\*

\* Class variable for the first argument

\*/

public static BigFraction arg1 = new BigFraction(BigInteger.ONE, BigInteger.ONE);

public static BigFraction answer = new BigFraction(BigInteger.ONE, BigInteger.ONE);

/\*\*

\* Class variable for the second argument

\*/

public static BigFraction arg2 = new BigFraction(BigInteger.ONE, BigInteger.ONE);

/\*\*

\* Buttons for numbers

\*/

private JButton zero = new JButton("0");

private JButton one = new JButton("1");

private JButton two = new JButton("2");

private JButton three = new JButton("3");

private JButton four = new JButton("4");

private JButton five = new JButton("5");

private JButton six = new JButton("6");

private JButton seven = new JButton("7");

private JButton eight = new JButton("8");

private JButton nine = new JButton("9");

/\*\*

\* buttons for common operations

\*/

private JButton adding\_button = new JButton("+");

private JButton subtraction\_button = new JButton("-");

private JButton multiplication\_button = new JButton("×");

private JButton dividing\_button = new JButton("÷");

private JButton answer\_button = new JButton("=");

/\*\*

\* buttons for non-standart operations

\*/

private JButton fruction\_button = new JButton("Fraction");

private JButton delete\_button = new JButton("⇐");

private JButton clear\_button = new JButton("C");

/\*\*

\* fields for input and output

\*/

private JLabel input\_field = new JLabel("");

private JTextField text\_field = new JTextField("", 1);

/\*\*

\* GridBagLayout

\*/

ActionListener actionListener = new TestActionListener();

GridBagLayout gridbag = new GridBagLayout();

GridBagConstraints c = new GridBagConstraints();

byte operation\_counter = 0; // + or - or / or \*

String label = " ";

/\*\*

\* Function for buttons

\*

\* @param gridwidth

\* @param weightx

\* @param gridx

\* @param gridy

\* @param button

\*/

public void make\_buttons(int gridwidth, double weightx, int gridx, int gridy, JButton button) {

c.gridwidth = gridwidth;

c.weightx = weightx;

c.gridx = gridx;

c.gridy = gridy;

gridbag.setConstraints(button, c);

add(button);

button.setFont(new Font("Serif", Font.BOLD, 30));

button.addActionListener(actionListener);

}

/\*\*

\* Function for action on operation button's press

\*

\* @param operation

\* @param line string

\*/

public void operations\_buttons(byte operation, String line, String arithmetic\_operation) {

if (fraction\_flag == true && line.length() != 0) {

label = label + line + arithmetic\_operation;

input\_field.setText(label);

operation\_counter = operation;

arg1.denominator = new BigInteger(text\_field.getText());

line = "";

text\_field.setText(line);

fraction\_flag = false;

}

}

public void end(String line) {

label = " ";

input\_field.setText(label);

operation\_counter = 0;

line = "";

text\_field.setText(line);

fraction\_counter = 0;

fraction\_flag = false;

end = false;

}

/\*\*

\* Constructor for main window

\*/

public BigFractionCalculatorGUI() {

super("Fractional calculator");

this.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

this.setSize(500, 800);

this.setLocationRelativeTo(null);

input\_field.setText(label);

setLayout(gridbag);

c.weighty = 1.0;

c.fill = GridBagConstraints.BOTH;

/\*\*

\* Input\_Field

\*/

c.gridwidth = GridBagConstraints.REMAINDER;

gridbag.setConstraints(input\_field, c);

add(input\_field);

input\_field.setComponentOrientation(ComponentOrientation.RIGHT\_TO\_LEFT);

input\_field.setFont(new Font("Serif", Font.BOLD, 20));

/\*\*

\* Text\_Field

\*/

gridbag.setConstraints(text\_field, c);

add(text\_field);

text\_field.setComponentOrientation(ComponentOrientation.RIGHT\_TO\_LEFT);

text\_field.setFont(new Font("Serif", Font.BOLD, 40));

/\*\*

\* Other buttons

\*/

make\_buttons(3, 1.0, 0, 2, fruction\_button);

make\_buttons(1, 0.2, 3, 2, delete\_button);

make\_buttons(1, 1.0, 0, 3, seven);

make\_buttons(1, 1.0, 1, 3, eight);

make\_buttons(1, 1.0, 2, 3, nine);

make\_buttons(1, 1.0, 3, 3, dividing\_button);

make\_buttons(1, 1.0, 0, 4, four);

make\_buttons(1, 1.0, 1, 4, five);

make\_buttons(1, 1.0, 2, 4, six);

make\_buttons(1, 1.0, 3, 4, multiplication\_button);

make\_buttons(1, 1.0, 0, 5, one);

make\_buttons(1, 1.0, 1, 5, two);

make\_buttons(1, 1.0, 2, 5, three);

make\_buttons(1, 1.0, 3, 5, subtraction\_button);

make\_buttons(1, 1.0, 0, 6, clear\_button);

make\_buttons(1, 1.0, 1, 6, zero);

make\_buttons(1, 1.0, 2, 6, answer\_button);

make\_buttons(1, 1.0, 3, 6, adding\_button);

}

byte fraction\_counter = 0;

boolean fraction\_flag = false;

boolean end = false;

public class TestActionListener implements ActionListener {

public void actionPerformed(ActionEvent e) {

String line = text\_field.getText();

try {

if (end == true) {

end(line);

}

if (e.getSource() == answer\_button) {

if (line.length() != 0 && fraction\_counter == 2) {

arg2.denominator = new BigInteger(text\_field.getText());

switch (operation\_counter) {

case 1:

answer = arg1.Addition(arg1);

break;

case 2:

answer = arg1.Subtraction(arg1);

break;

case 3:

answer = arg1.Multiplication(arg1);

break;

case 4:

answer = arg1.Division(arg1);

break;

}

label = label + line + " = " + answer.numerator + "/" + answer.denominator;

input\_field.setText(label);

line = "";

text\_field.setText(line);

end = true;

}

} else if (e.getSource() == clear\_button) {

end(line);

} else if (e.getSource() == delete\_button) {

if (line.length() != 0)

line = line.substring(0, line.length() - 1);

text\_field.setText(line);

} else if (e.getSource() == fruction\_button) {

if (line.length() != 0 && fraction\_flag == false && fraction\_counter != 2) {

fraction\_flag = true;

line = text\_field.getText();

label = label + line + "/";

input\_field.setText(label);

if (operation\_counter == 0) {

arg1.numerator = new BigInteger(text\_field.getText());

fraction\_counter = 1;

} else {

arg2.numerator = new BigInteger(text\_field.getText());

fraction\_flag = false;

fraction\_counter = 2;

}

line = "";

text\_field.setText(line);

}

} else if (e.getSource() == adding\_button) {

operations\_buttons((byte) 1, line, " + ");

} else if (e.getSource() == subtraction\_button) {

operations\_buttons((byte) 2, line, " - ");

} else if (e.getSource() == multiplication\_button) {

operations\_buttons((byte) 3, line, " × ");

} else if (e.getSource() == dividing\_button) {

operations\_buttons((byte) 4, line, " ÷ ");

} else {

if (e.getSource() == zero && line.length() != 0) {

text\_field.setText(line + "0");

} else if (e.getSource() == one) {

text\_field.setText(line + "1");

} else if (e.getSource() == two) {

text\_field.setText(line + "2");

} else if (e.getSource() == three) {

text\_field.setText(line + "3");

} else if (e.getSource() == four) {

text\_field.setText(line + "4");

} else if (e.getSource() == five) {

text\_field.setText(line + "5");

} else if (e.getSource() == six) {

text\_field.setText(line + "6");

} else if (e.getSource() == seven) {

text\_field.setText(line + "7");

} else if (e.getSource() == eight) {

text\_field.setText(line + "8");

} else if (e.getSource() == nine) {

text\_field.setText(line + "9");

}

}

} catch (Exception ee) {

end(line);

}

}

}

}

Приложение Б

(обязательно)

Экранные формы

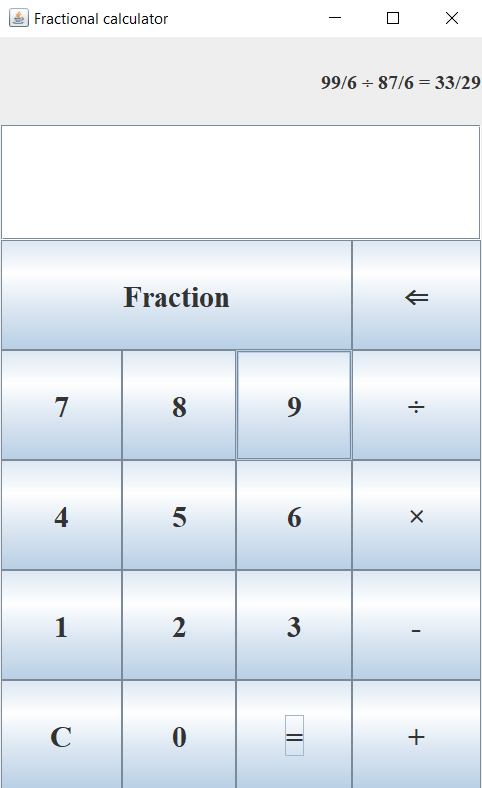


Рисунок 1 – Пользовательский интерфейс дробного калькулятора

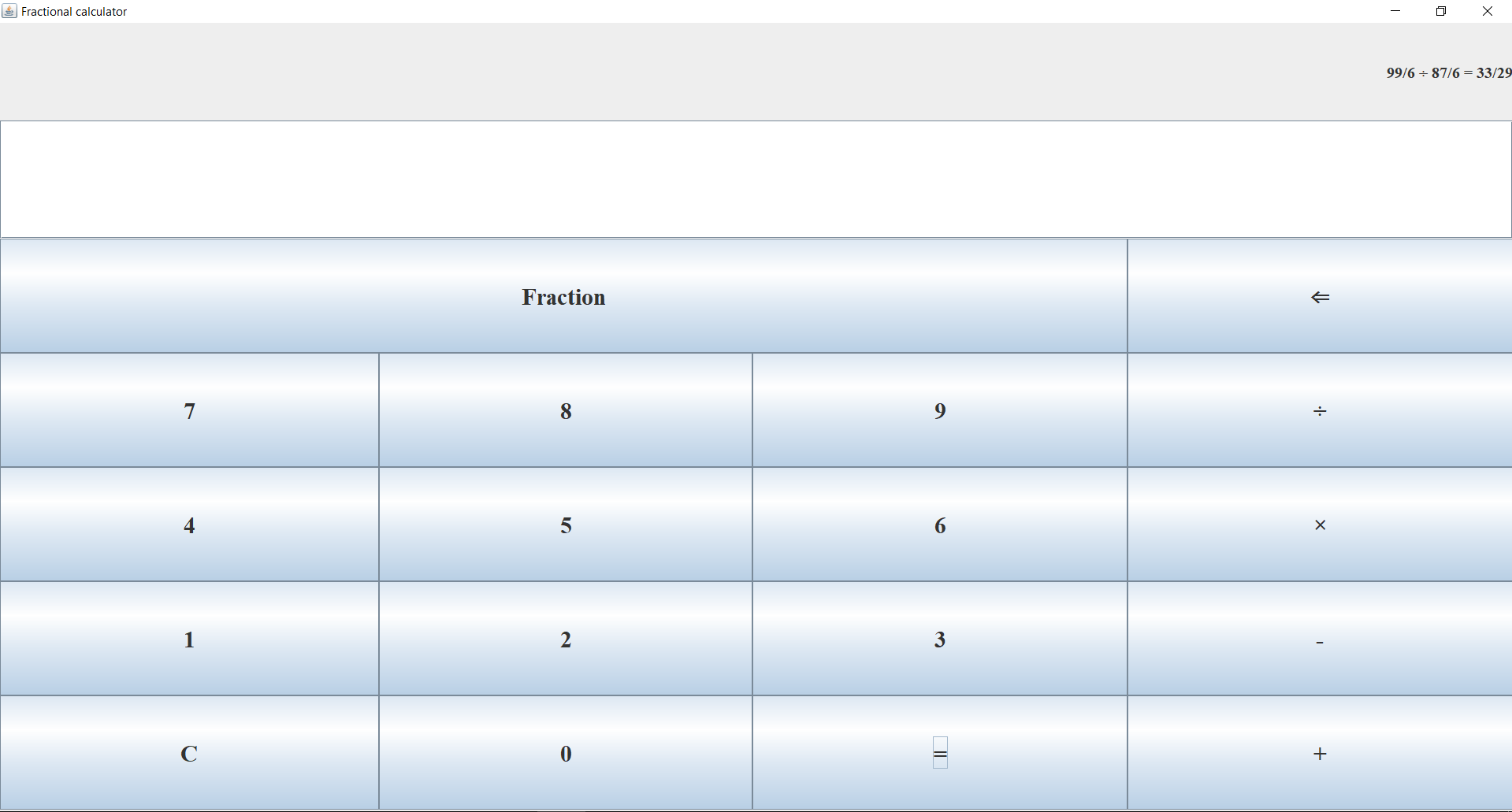


Рисунок 2 – Калькулятор в развёрнутом виде