**Лабораторная работа №3**

**«Разветвляющийся алгоритм»**

**1.Задание на разработку проекта**

Создать приложение под названием «Лабораторная 3» для вычисления максимальных и минимальных значений функций k = f(x,y,a,b) при определенных значениях x и у.

Задано: x, y, a, b – переменные

Вычислить функцию k = f(x,y,a,b):

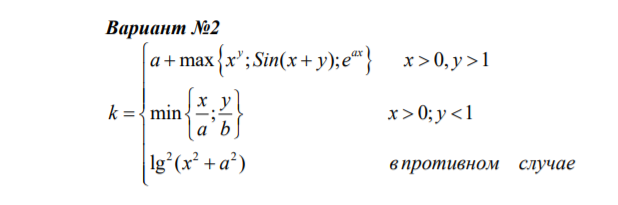


Рисунок 1 – функция k = f(x,y,a,b)

Ограничения на вводимые данные:

1. При x > 0 и y < 1:

a и b не могут равняться нулю

1. При x > 1 и y < 0:

Ограничений нет

1. В остальных случаях:

x и а одновременно не должны равняться нулю

**2. Формализация и решения задачи**

Алгоритм решения данной задачи представляет собой комбинацию вычисления сложной функции с условием выбора наибольшего из нескольких значений. Данная задача решается с помощью метода **Razv()**, который, получив а качестве входных параметров аргументы x, y, a, b, возвращает вычисленное значение.

В соответствии с требованиями задания разобьем решение задачи на несколько отдельных задач и создадим подпрограммы пользователя, реализующие эти задачи:

* Метод double **Vvod( )**,созданный для ввода значений с ***TextBox;***
* Метод void **Vivod()**, созданный для вывода значений в ***TextBox;***
* Метод string **Prof()**, созданный для проверки вводимых значений;
* Метод string **Razv()**, созданный для вычисления значений;

**3. Разработка проекта**

3.1. Разработка графического интерфейса пользователя

Разработанная форма интерфейса пользователя приведена на рис. 2.

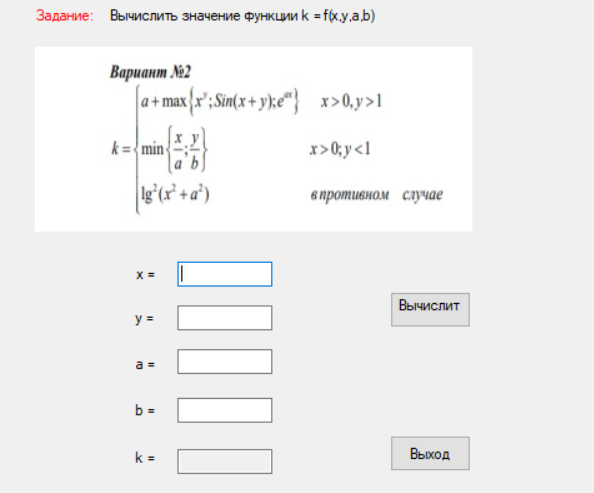


Рисунок 2 – интерфейс для пользователя

3.2. Разработка схемы алгоритма

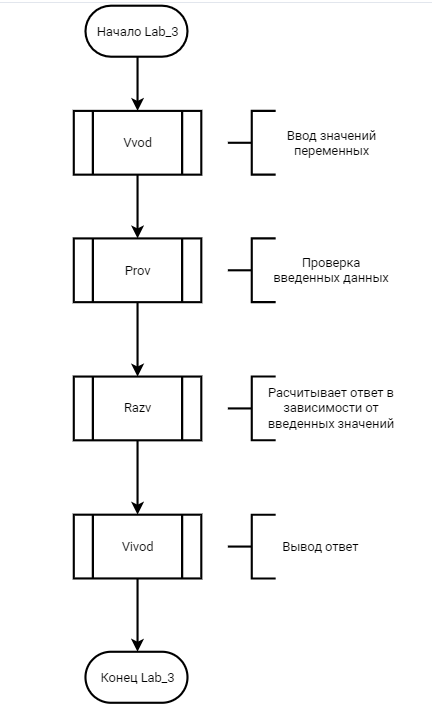


Рисунок 3 – Схема алгоритма процедур

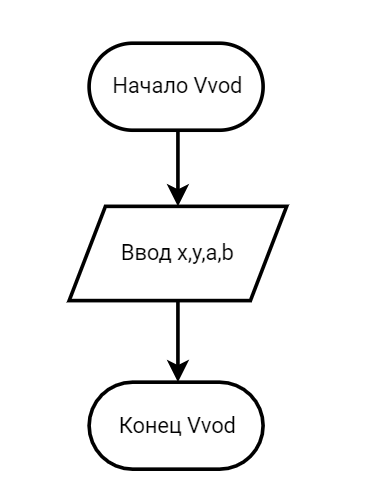


Рисунок 4 – Схема алгоритмов процедур Vvod

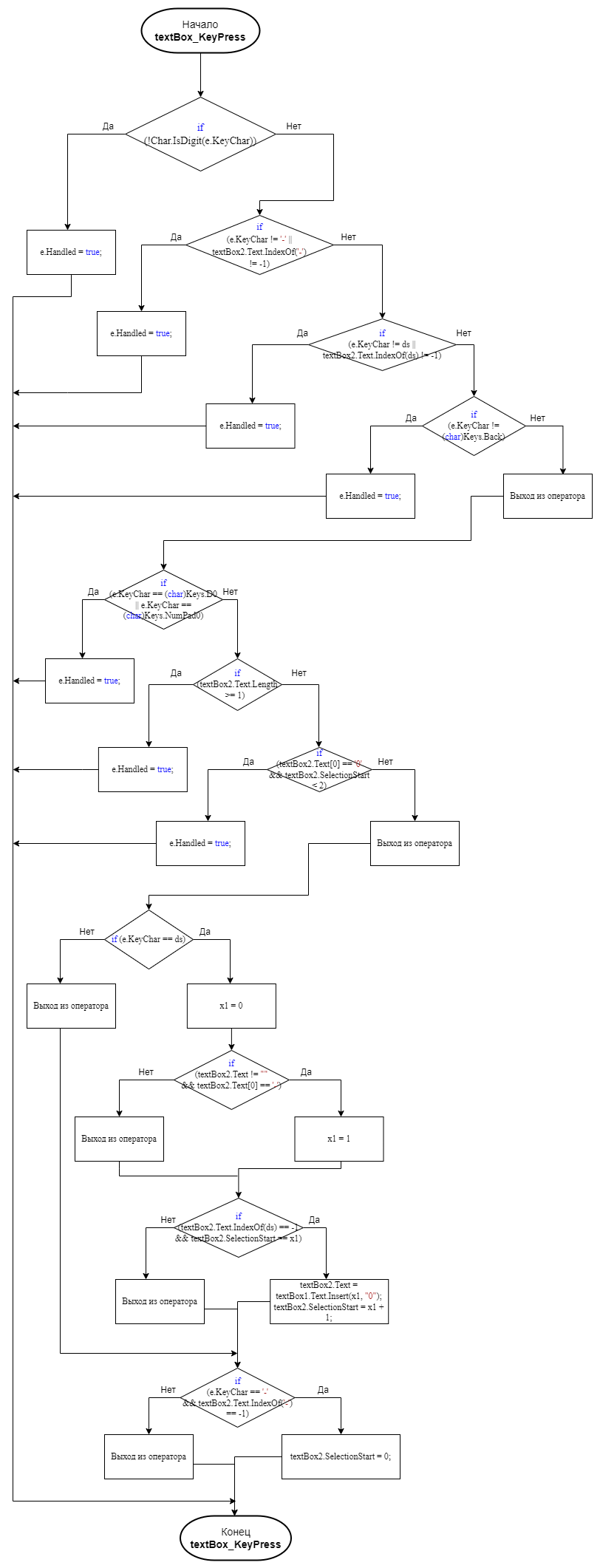


Рисунок 5 - Защита от дурака

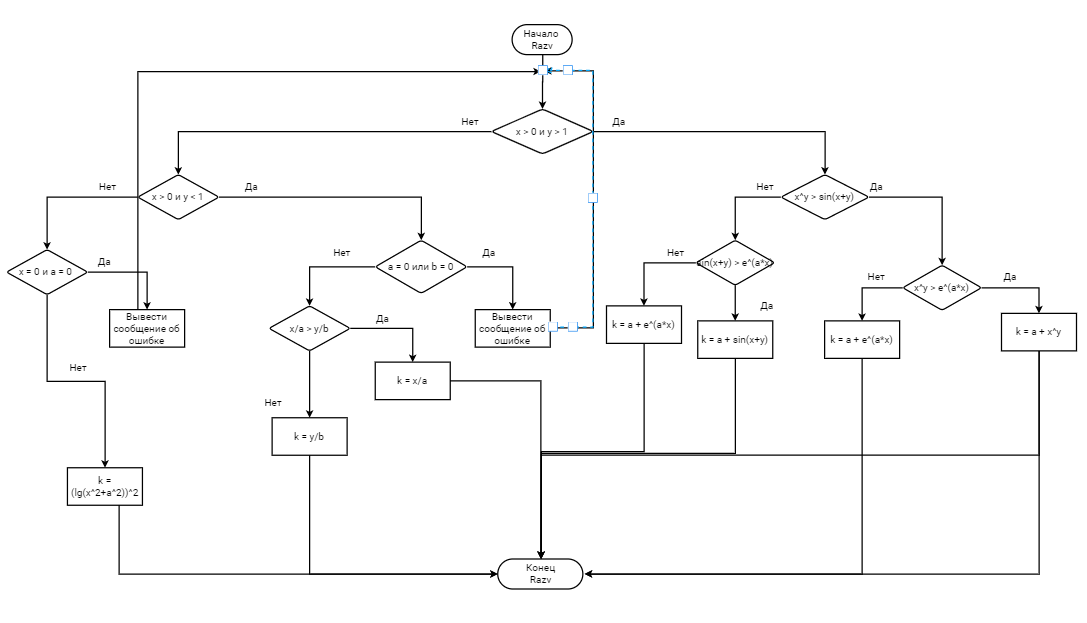


Рисунок 6 – Схема алгоритмов процедур Razv

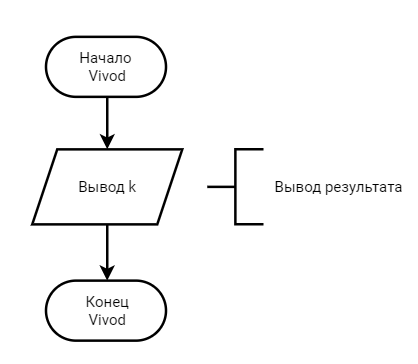


Рисунок 7 – Схема алгоритмов процедур Vivod

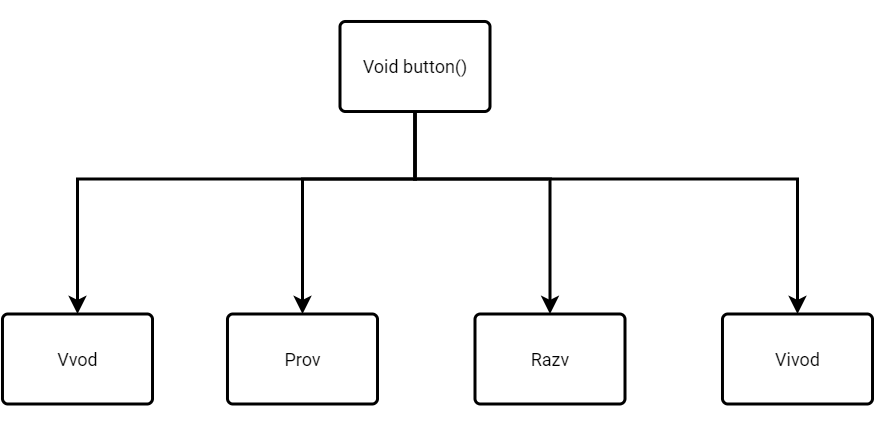


Рисунок 8 – Схема иерархии процедур для решения задач

3.3. Разработка программного кода

Все подпрограммы-методы размещаются в классе. У меня этим классом выступает Class3

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Globalization;

namespace \_3Laba

{

public class Class3

{

// метод вводa в TextBox

public static double Vvod(TextBox textbox)

{

return Convert.ToDouble(textbox.Text);

}

// Вывод в TextBox

public static void Vivod(TextBox textBox, double c)

{

textBox.Text = Convert.ToString(c);

}

public static string Prof(TextBox textBox, KeyPressEventArgs e)

{

int x1;

char ds = (char)NumberFormatInfo.CurrentInfo.NumberDecimalSeparator[0];

if (!Char.IsDigit(e.KeyChar))

// Запрет на ввод более одного десятичного разделителя.

if (e.KeyChar != ds || textBox.Text.IndexOf(ds) != -1)

// Если нажатая клавиша не является клавишей BackSpace.

if (e.KeyChar != (char)Keys.Back)

e.Handled = true; // Запрет ввода

// Запрет ввода повторных нулей в начале числа.

if (e.KeyChar == '-')

e.Handled = true;

if (e.KeyChar == (char)Keys.D0 || e.KeyChar == (char)Keys.NumPad0)

if (textBox.Text.Length >= 1) // Если строка не пустая

if (textBox.Text[0] == '0' && textBox.SelectionStart < 2)

e.Handled = true; // Запрет ввода

//-незначащий ноль

if (Char.IsDigit(e.KeyChar))

if (textBox.Text.Length == 1 && textBox.Text[0] == '0')

textBox.Paste(",");

// Замена десятичного разделителя в начале числа на "0,".

if (e.KeyChar == ds)

{ // Если введен десятичный разделитель

x1 = 0;

if (textBox.Text != "" && textBox.Text[0] == '-') { x1 = 1; }

if (textBox.Text.IndexOf(ds) == -1 && textBox.SelectionStart == x1)

{

// Если десятичный разделитель уже есть, то 0 не вставлять.

textBox.Text = textBox.Text.Insert(x1, "0"); // Вставка 0

textBox.SelectionStart = x1 + 1; // Текстовый курсор в конец

}

// Перенос точки ввода минуса в начало (ввод минуса только спереди)

}

if (Char.IsDigit(e.KeyChar))

if (textBox.Text.Length == 1 && textBox.Text[0] == '0')

textBox.Paste(",");

return "";

}

public static string Razv(double x, double y, double a, double b, out double k)

{

k = 0;

if ((x > 0) && (y > 1))

{

if (Math.Pow(x, y) > Math.Sin(x + y))

{

if (Math.Pow(x, y) > Math.Pow(Math.Exp(1), a \* x))

{

k = a + Math.Pow(x, y);

}

else

{

k = a + Math.Pow(Math.Exp(1), a \* x);

}

}

else

{

if (Math.Sin(x + y) > Math.Pow(Math.Exp(1), a \* x))

{

k = a + Math.Sin(x + y);

}

else

{

k = a + Math.Pow(Math.Exp(1), a \* x);

}

}

}

else if ((x > 0) && (y < 1))

{

if (x / a > y / b)

{

if (a != 0)

k = x / a;

else

{

MessageBox.Show("В данном случае 'а' не должно равняться нулю");

}

}

else

{

if (b != 0)

k = y / b;

else

{

MessageBox.Show("В данном случае 'b' не должно равняться нулю");

}

}

}

else

{

if (x != 0 || a != 0)

k = Math.Pow(Math.Log10(Math.Pow(x, 2) + Math.Pow(a, 2)), 2);

else

{

MessageBox.Show("В данном случае 'x' совместно с 'a' не должно равняться нулю");

k = 0;

}

}

return Convert.ToString(k);

}

}

}

В основной части проекта подпрограммы-методы вызываются из Class3

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.Globalization;

namespace AnP\_proj

{

public partial class Form4 : Form

{

public Form4()

{

InitializeComponent();

}

private void label5\_Click(object sender, EventArgs e)

{

}

private void textBox1\_TextChanged(object sender, EventArgs e)

{

}

private void button1\_Click(object sender, EventArgs e)

{

double x, y, a, b, k;

x = \_3Laba.Class3.Vvod(textBox1);

y = \_3Laba.Class3.Vvod(textBox2);

a = \_3Laba.Class3.Vvod(textBox3);

b = \_3Laba.Class3.Vvod(textBox4);

\_3Laba.Class3.Razv(x, y, a, b, out k);

\_3Laba.Class3.Vivod(textBox5, k);

}

private void button2\_Click(object sender, EventArgs e)

{

Application.Exit();

}

private void False(object sender, EventArgs e)

{

}

private void textBox1\_KeyPress(object sender, KeyPressEventArgs e)

{

\_3Laba.Class3.Prof(textBox1, e);

}

private void textBox2\_KeyPress(object sender, KeyPressEventArgs e)

{

\_3Laba.Class3.Prof(textBox2, e);

}

private void textBox3\_KeyPress(object sender, KeyPressEventArgs e)

{

\_3Laba.Class3.Prof(textBox3, e);

}

private void textBox4\_KeyPress(object sender, KeyPressEventArgs e)

{

\_3Laba.Class3.Prof(textBox4, e);

}

}

}

**4. Выполнение приложения и проверка результатов**

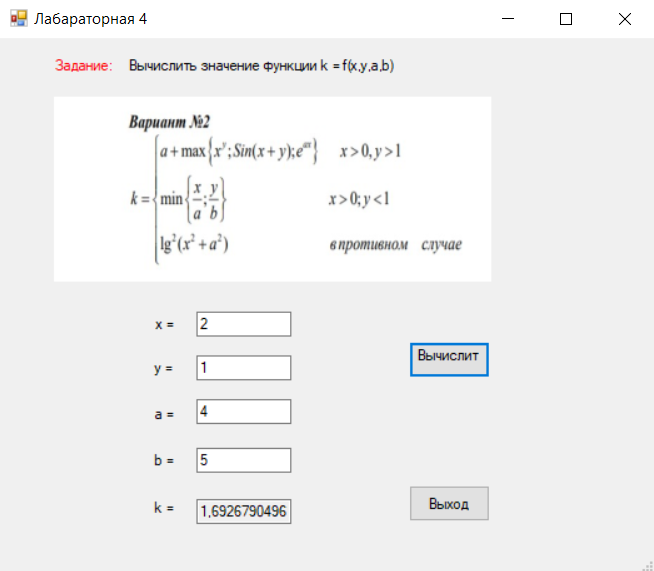
****

Рисунок 9 – Проверка работы программы

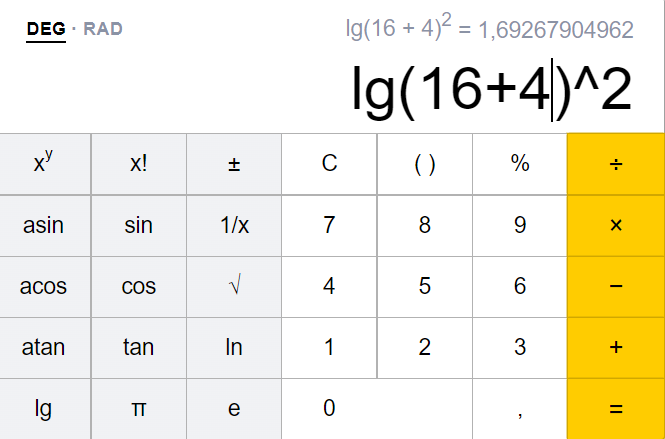


Рисунок 10 – Проверка работы программы

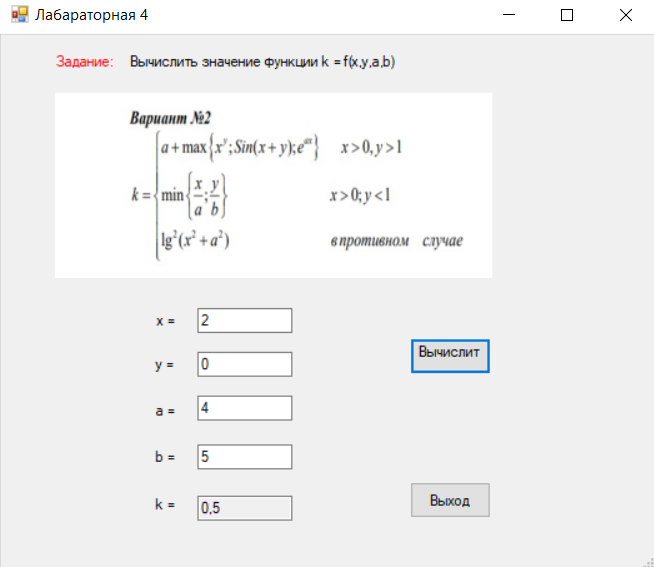


Рисунок 11 – Проверка работы программы

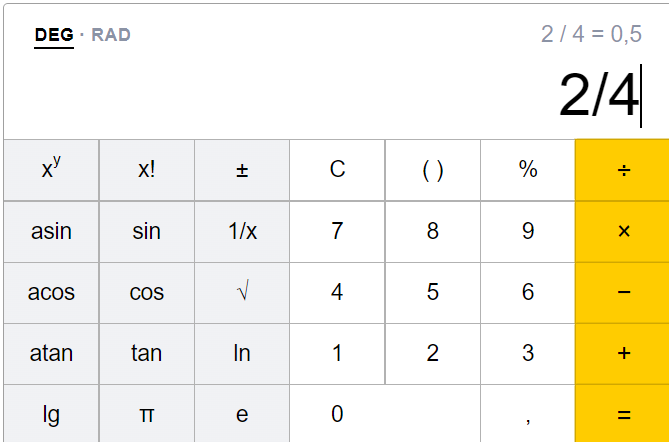


Рисунок 12 – Проверка работы программы

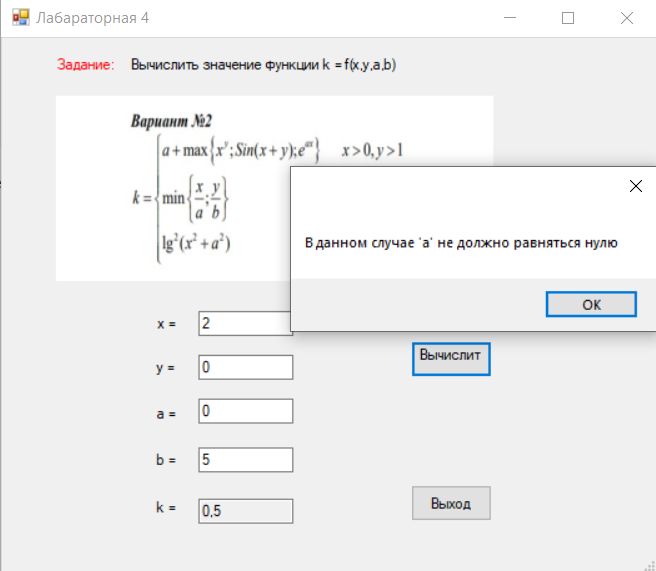


Рисунок 13 – Проверка работы программы