

**Министерство науки и высшего образования**

**Российской Федерации**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО «МГТУ «СТАНКИН»)**

Институт цифровых интеллектуальных систем

Дисциплина: «Программное обеспечение мехатронных систем»

Лабораторная работа № 1

Вариант 6

Выполнил:

студент группы АДМ-21-05 \_\_\_\_\_\_\_\_\_\_ Абдулзагиров М.М.

(подпись) (ФИО)

Принял

преподаватель: \_\_\_\_\_\_\_\_\_\_ \_ Поливанов А.Ю. \_\_

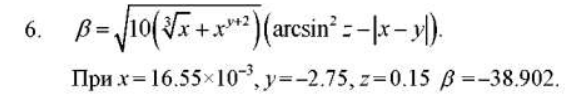
(подпись) (ФИО)

Дата:\_\_\_\_\_\_\_\_\_

Москва 2022

# Задание 1. Линейные алгоритмы

**Цель задания**: научиться составлять каркас простейшей программы в среде Visual Studio. Написать и отладить программу линейного алгоритма.



# Описание программы

Исходный код программы:

Листинг 1. Файл Form1.cs

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 static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace Lab\_1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

// Начальное значение X

textBox1.Text = "0.01655";

// Начальное значение Y

textBox2.Text = "-2.75";

// Начальное значение Z

textBox3.Text = "0.15";

}

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

{

try

{

// Считывание значения X

double x = double.Parse(textBox1.Text.Replace('.',','));

// Вывод значения X в окно

textBox4.Text = Environment.NewLine +

"X = " + x.ToString();

// Считывание значения Y

double y = double.Parse(textBox2.Text.Replace('.', ','));

// Вывод значения Y в окно

textBox4.Text += Environment.NewLine +

"Y = " + y.ToString();

// Считывание значения Z

double z = double.Parse(textBox3.Text.Replace('.', ','));

// Вывод значения Z в окно

textBox4.Text += Environment.NewLine +

"Z = " + z.ToString();

double u = SixTask(x, y, z);

// Выводим результат в окно

textBox4.Text += Environment.NewLine +

"\nРезультат U = " + u.ToString();

}

catch (Exception ex)

{

MessageBox.Show("Введены некорректыне данные\n" + ex.Message,

"Ошибка",

MessageBoxButtons.YesNo,

MessageBoxIcon.Error);

// textBox4.Text = "Введены некорректыне данные";

}

}

double SixTask(double x, double y, double z)

{

double t1 = Math.Sqrt(10.0 \* (Math.Pow(x, 1.0 / 3.0) + Math.Pow(x, y + 2.0)));

double t2 = (Math.Pow(Math.Asin(z - Math.Abs(x - y)), 2) );

return t1\*t2;

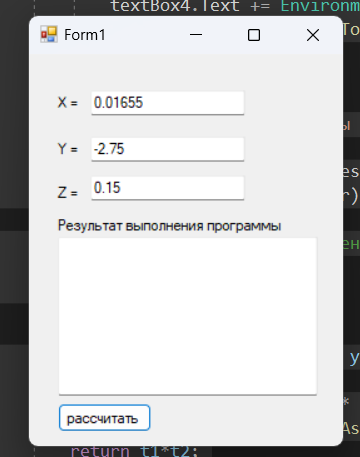
}

}

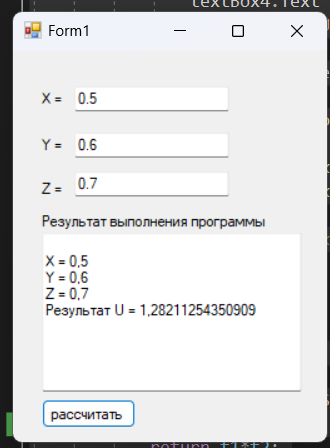
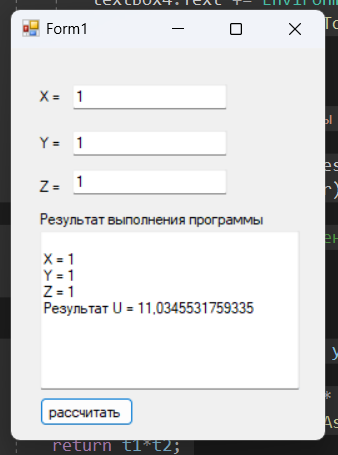
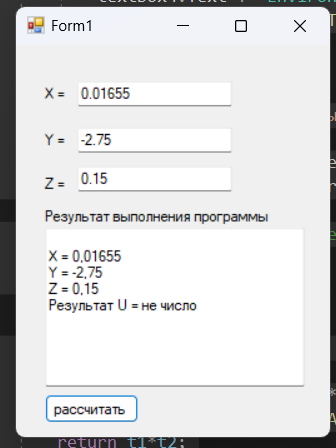
}

# Результаты выполнения программы

Запустим программу



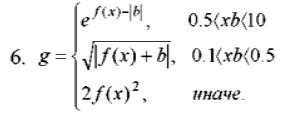
По заданным данным ответ не рассчитывается (комплексное число) т.к. при выходе за пределы функция arcsin выдаёт комплексное число, в данном случа просто Nan.



**Вывод**: в данном задании мы научились составлять каркас простейшей программы в среде Visual Studio, а также написали и отладили программу линейного алгоритма.

# Задание 2. Разветвляющиеся алгоритмы

**Цель задания**: научиться пользоваться элементами управления для организации переключений (RadioButton). Написать и отладить программу разветвляющегося алгоритма.



# Описание программы

Исходный код программы:

Листинг 1. Файл Form1.cs

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;

namespace Lab\_1\_Task\_2

{

public partial class Form1 : Form

{

const double X\_0 = 3.4, Y\_0 = 0.74, Z\_0 = 19.43;

FunctionType fType;

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

// Начальное значение X

textBox1.Text = X\_0.ToString();

// Начальное значение Y

textBox2.Text = Y\_0.ToString();

// Начальное значение Z

textBox3.Text = Z\_0.ToString();

radioButton1.Checked = true;

}

private void radioButton\_CheckedChanged(object sender, EventArgs e)

{

if (sender.Equals(radioButton1))

fType = FunctionType.sh\_x;

else if (sender.Equals(radioButton2))

fType = FunctionType.x2;

else

fType = FunctionType.e\_x;

}

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

{

try

{

// Считывание значения X

double x = double.Parse(textBox1.Text.Replace('.',','));

// Вывод значения X в окно

textBox4.Text = Environment.NewLine +

"X = " + x.ToString();

// Считывание значения Y

double y = double.Parse(textBox2.Text.Replace('.', ','));

// Вывод значения Y в окно

textBox4.Text += Environment.NewLine +

"Y = " + y.ToString();

// Считывание значения Z

double z = double.Parse(textBox3.Text.Replace('.', ','));

// Вывод значения Z в окно

textBox4.Text += Environment.NewLine +

"Z = " + z.ToString();

// Вычисляем арифметическое выражение

double u = SixTask2(x, y);

// Выводим результат в окно

textBox4.Text += Environment.NewLine +

"\nРезультат U = " + u.ToString();

}

catch (Exception ex)

{

MessageBox.Show("Введены некорректыне данные\n" + ex.Message,

"Ошибка",

MessageBoxButtons.YesNo,

MessageBoxIcon.Error);

// textBox4.Text = "Введены некорректыне данные";

}

}

double SixTask2(double x, double b)

{

double g;

if (x \* b > 0.5 && x \* b < 10)

g = Math.Pow(Math.E, f(x) - Math.Abs(b));

if (x \* b < 0.5 && x \* b > 0.1)

g = Math.Sqrt(Math.Abs(f(x)+b));

else

g = 2.0 \* Math.Pow(f(x),2.0);

return g;

}

double f(double x)

{

switch (fType)

{

case FunctionType.sh\_x:

return Math.Sinh(x);

case FunctionType.x2:

return Math.Pow(x, 2.0);

case FunctionType.e\_x:

return Math.Pow(Math.E, x);

default:

throw new Exception("Неверный тип уравнения");

}

}

enum FunctionType

{

sh\_x, x2, e\_x

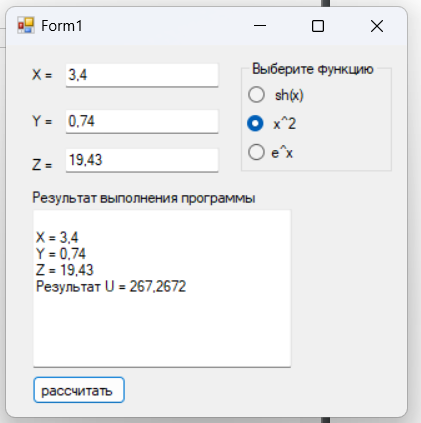
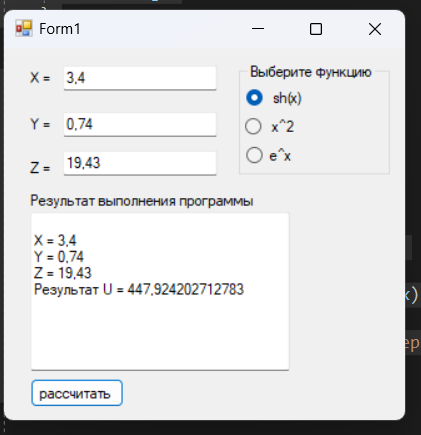
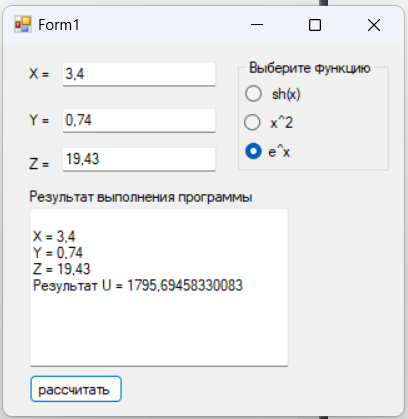
}

}

}

# Результаты выполнения программы

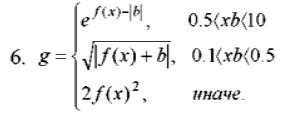
Запустим программу. В зависимости от выбранной функции изменяется и рассчитываемое значение.

**Вывод**: в данном задании мы научились пользоваться элементами управления для организации переключений (RadioButton), а также написали и отладили программу разветвляющегося алгоритма.

# Задание 3. Графики функций

**Цель задания**: изучить возможности построения графиков с помощью элемента управления Сhart. Написать и отладить программу построения на экране графика заданной функции.



# Описание программы

Исходный код программы:

Листинг 1. Файл Form1.cs

using System;

using System.Windows.Forms;

namespace Lab\_1\_3

{

public partial class Form1 : Form

{

const double x\_0 = -2.5, x\_k = -1.5, d\_x = 0.5;

const double b = -0.8;

public Form1()

{

InitializeComponent();

}

private void Form1\_Load(object sender, EventArgs e)

{

textBoxXmin.Text = x\_0.ToString();

textBoxXmax.Text = x\_k.ToString();

textBoxStep.Text = d\_x.ToString();

}

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

{

// Считываем с формы требуемые значения

double Xmin = double.Parse(textBoxXmin.Text);

double Xmax = double.Parse(textBoxXmax.Text);

double Step = Math.Abs (double.Parse(textBoxStep.Text));

if (Xmin > Xmax) // меняем местами

{

Xmin+=Xmax;

Xmax= -1.0\*(Xmax - Xmin);

Xmin -= Xmax;

}

// Количество точек графика

int count = (int) Math.Abs(Math.Ceiling((Xmax - Xmin) / Step) + 1);

// Массив значений X – общий для обоих графиков

double[] x = new double[count];

// Два массива Y – по одному для каждого графика

double[] y1 = new double[count];

double[] y2 = new double[count];

// Расчитываем точки для графиков функции

for (int i = 0; i < count; i++)

{

// Вычисляем значение X

x[i] = Xmin + Step \* i;

// Вычисляем значение функций в точке X

y1[i] = f1(x[i]);

y2[i] = f2(x[i]);

}

// Настраиваем оси графика

chart1.ChartAreas[0].AxisX.Minimum = Xmin;

chart1.ChartAreas[0].AxisX.Maximum = Xmax;

// Определяем шаг сетки

chart1.ChartAreas[0].AxisX.MajorGrid.Interval = Math.Abs(Step);

// Добавляем вычисленные значения в графики

chart1.Series[0].Points.DataBindXY(x, y1);

chart1.Series[1].Points.DataBindXY(x, y2);

}

double f1(double x)

{

return Math.Pow(x, 2) + Math.Tan(5.0\*x+ b/x);

}

double f2(double x)

{

return 6.0\*Math.Cos(5.0 \* x);

//return -0.6 \* Math.Pow(x, 3) + 2.0 \*Math.Pow(x,2) + 5.0\*x + x\_0 ;

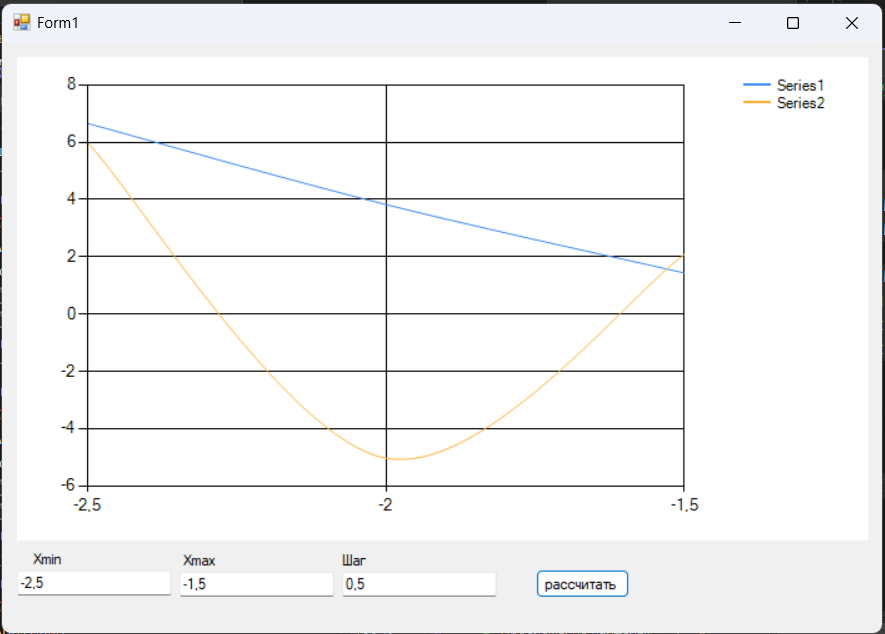
}

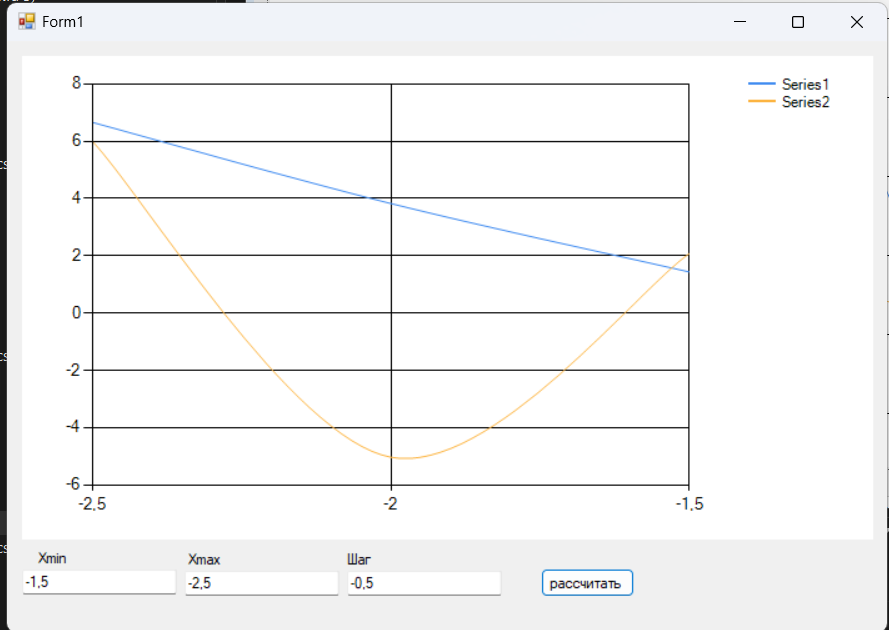
}

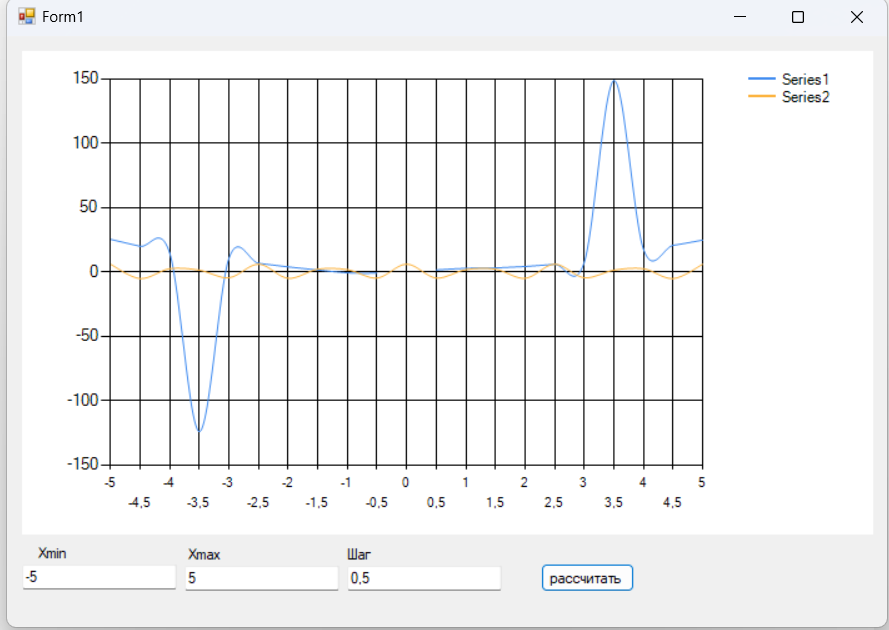
}

# Результаты выполнения программы

Запустим программу. В зависимости от введённых данных изменяются графики. При вводе отрицательного шага в данном случае используется его модуль. При вводе диапазона, при котором Xmin > Xmax график выводится с направлением оси Х также слева на право (не удалось обработать ошибку для отображения в обратном направлении).







**Вывод**: в данном задании мы изучили возможности построения графиков с помощью элемента управления Сhart, а также написали и отладили программу построения на экране графика заданной функции.

Ссылка на файл с исходным кодом

https://disk.yandex.ru/d/Eh8cpDxJMUzcPA