**Лабораторная работа №5**

**“Циклические алгоритмические структуры. Регулярные циклические структуры и их программная реализация”**

1. **Постановка задачи**

Постройте таблица значений функции **z = f(x, y)** и найдите сумму и количество положительных значение функции при изменении **x** на отрезке **[a; b]** с шагом **h1** и **y** на отрезке **[c; d]** и с шагом **h2**.

(Например: a = -3, b = 0, c = -3, d = 3, h1 = 0.3, h2 = 0.6)

(1)

Также требуется выгрузить данные в отдельный файл.

Для выполнения я использовал методы, представленные ниже:

* Метод ***Vvod()***, предназначенный для ввода данных;
* Метод ***Vivod()***, предназначенный для вывода данных;
* Метод ***VivodDGV()***, предназначенный для вывода данных на каждом шаге;
* Метод ***Func()***, который вычисляет исходную функцию;
* Метод ***Tab()***, который изменяет x и y на отрезке с определённым шагом;
* Метод ***DGVfile(DataGridView gridView)*** для выгрузки данных из таблицы в отдельный файл.

1. **Разработка проекта**
   1. Разработка графического интерфейса пользователя

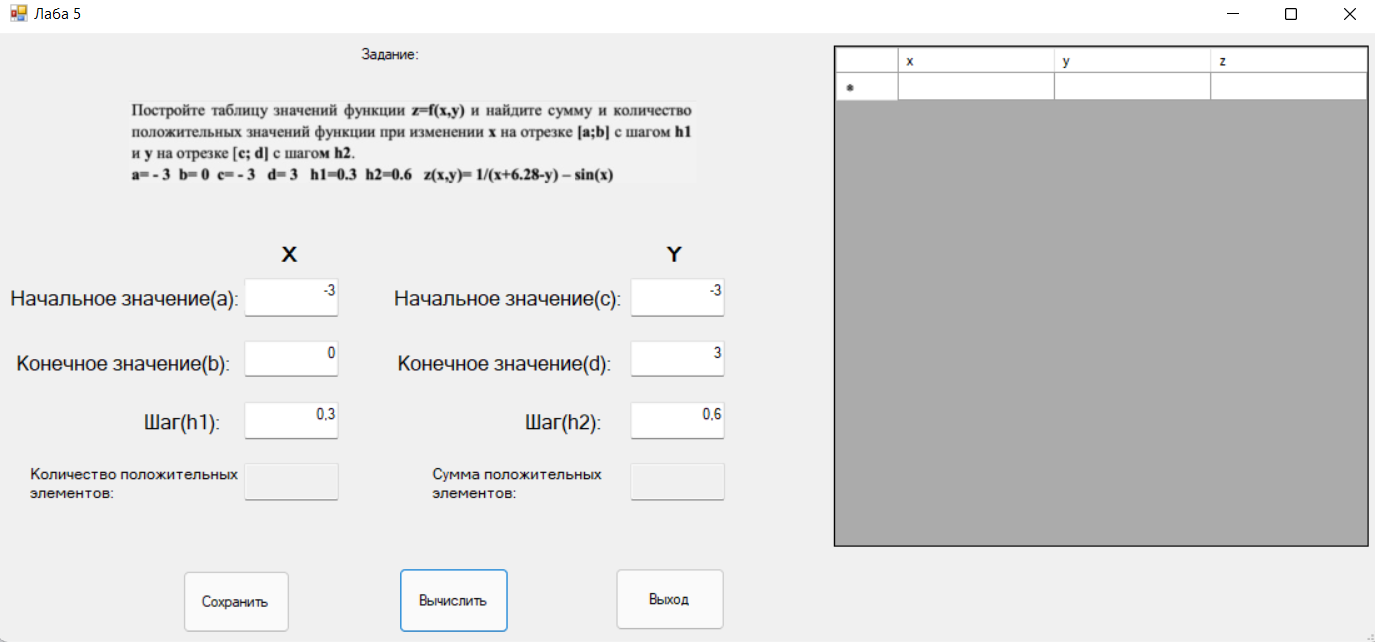
****

Рисунок 1 - Графический интерфейс программы

2.2. Разработка блок-схемы алгоритмов

Схемы событийной части проекта предоставлены на рисунках 2 - 4.

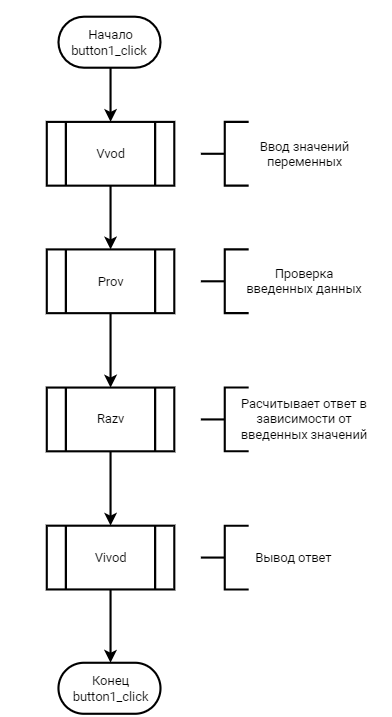


Рисунок 2 - Схема алгоритмов процедур **button\_Click1**

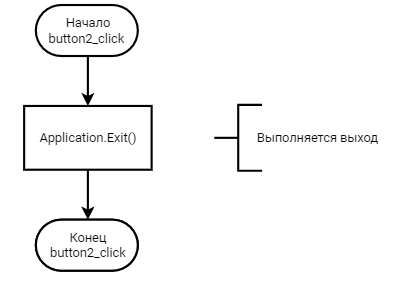


Рисунок 3 - Схема алгоритмов процедур **button\_Click2**

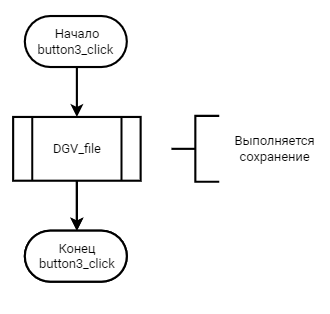


Рисунок 4 - Схема алгоритмов процедур **button\_Click2**

Блок схемы, подключенные из библиотек, представлены на рисунках 5-9.

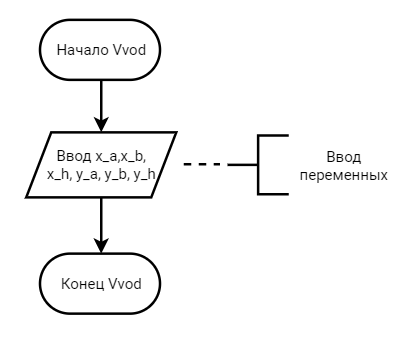


Рисунок 5 - Схема алгоритмов процедур **Vvod**

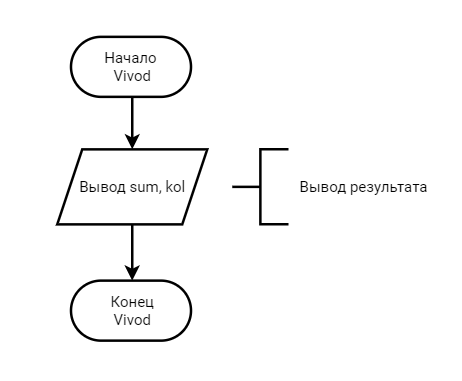


Рисунок 6 - Схема алгоритмов процедур **Vivod**

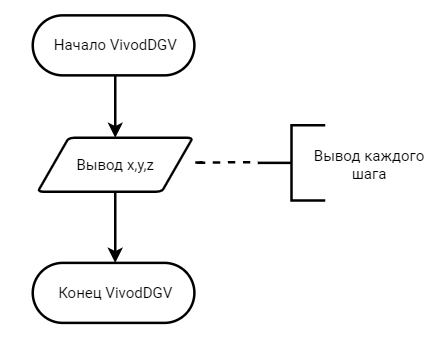


Рисунок 6 - Схема алгоритмов процедур **VivodDGV**

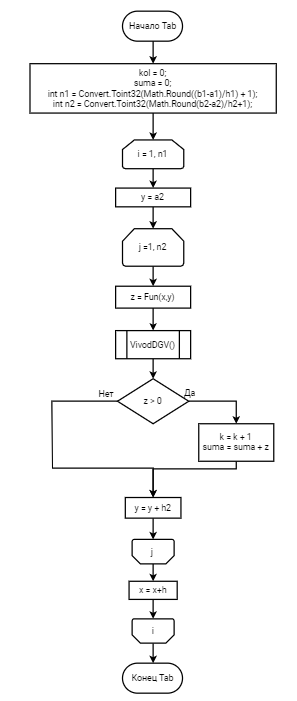
****

Рисунок 8 - Схема алгоритмов процедур **Tab**

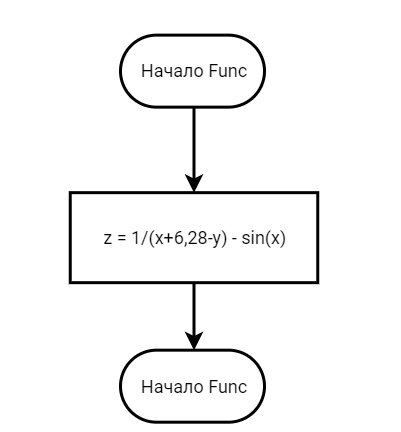
****

Рисунок 9 - Схема алгоритмов процедур **Func**

**3. Написание программного кода**

Код из основной части проекта:

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 AnP\_proj

{

public partial class Form6 : Form

{

public Form6()

{

InitializeComponent();

}

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

{

Application.Exit();

}

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

{

if (textBox1.Text == "" || textBox2.Text == "" || textBox3.Text == "" ||

textBox4.Text == "" || textBox5.Text == "" || textBox6.Text == "")

{

DialogResult = MessageBox.Show(this,

"Введены пустые значения!",

"Лабораторная работа №5",

MessageBoxButtons.OKCancel,

MessageBoxIcon.Exclamation,

MessageBoxDefaultButton.Button1);

return;

}

dataGridView1.Rows.Clear();

double x\_a, x\_b, x\_h, y\_a, y\_b, y\_h;

x\_a = Laba\_5.Class1.Vvod(textBox1);

x\_b = Laba\_5.Class1.Vvod(textBox2);

x\_h = Laba\_5.Class1.Vvod(textBox3);

y\_a = Laba\_5.Class1.Vvod(textBox4);

y\_b = Laba\_5.Class1.Vvod(textBox5);

y\_h = Laba\_5.Class1.Vvod(textBox6);

if (!(x\_h > 0) || !(y\_h > 0))

{

DialogResult = MessageBox.Show(this,

"Шаг должен быть больше '0'!",

"Лабораторная работа №5",

MessageBoxButtons.OKCancel,

MessageBoxIcon.Exclamation,

MessageBoxDefaultButton.Button1);

return;

}

if (!(x\_b > x\_a) || (x\_b == x\_a))

{

textBox1.Text = Convert.ToString(x\_b);

textBox2.Text = Convert.ToString(x\_a);

DialogResult = MessageBox.Show(this,

"Конечное значение должно быть больше начального!",

"Лабораторная работа №5",

MessageBoxButtons.OKCancel,

MessageBoxIcon.Exclamation,

MessageBoxDefaultButton.Button1);

return;

}

if (x\_h > (x\_b - x\_a) || y\_h > (y\_b - y\_a))

{

DialogResult = MessageBox.Show(this,

"Шаг не должен быть больше разности конечного и началального значения!",

"Лабораторная работа №5",

MessageBoxButtons.OKCancel,

MessageBoxIcon.Exclamation,

MessageBoxDefaultButton.Button1);

return;

}

if (!(y\_b > y\_a) || (y\_b == y\_a))

{

textBox4.Text = Convert.ToString(y\_b);

textBox5.Text = Convert.ToString(y\_a);

DialogResult = MessageBox.Show(this,

"Конечное значение должно быть больше начального!",

"Лабораторная работа №5",

MessageBoxButtons.OKCancel,

MessageBoxIcon.Exclamation,

MessageBoxDefaultButton.Button1);

return;

}

Laba\_5.Class1.Tab(x\_a, x\_b, x\_h, y\_a, y\_b, y\_h, dataGridView1, out double suma, out double kol);

Laba\_5.Class1.Vivod(textBox7, kol);

Laba\_5.Class1.Vivod(textBox8, suma);

}

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

{

Laba\_5.Class1.Prof(textBox1,e);

}

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

{

Laba\_5.Class1.Prof(textBox2,e);

}

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

{

Laba\_5.Class1.Prof(textBox3,e);

}

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

{

Laba\_5.Class1.Prof(textBox4,e);

}

private void textBox5\_KeyPress(object sender, KeyPressEventArgs e)

{

Laba\_5.Class1.Prof(textBox5,e);

}

private void textBox6\_KeyPress(object sender, KeyPressEventArgs e)

{

Laba\_5.Class1.Prof(textBox6,e);

}

private void button3\_Click(object sender, EventArgs e)

{

Laba\_5.Class1.Zapis(dataGridView1, textBox1, textBox2, textBox3, textBox4, textBox5, textBox6, textBox7);

}

}

}

Код из библиотеки Class1:

using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

using System.IO;

using System.Globalization;

using Microsoft.VisualBasic;

using System.Diagnostics;

namespace Laba\_5

{

public class Class1

{

public static double Vvod(TextBox textbox)

{

return Convert.ToDouble(textbox.Text);

}

public static void Vivod(TextBox t, double c)

{

t.Text = c.ToString("F");

}

public static double Func(double x, double y)

{

double z = 1/(x+6.28-y) - Math.Sin(x);

return z;

}

public static void VivodDGV(double x, double y, double z, DataGridView DGV1)

{

DGV1.Rows.Add(x.ToString("F1"), y.ToString("F5"), z.ToString("F5"));

}

public static void Tab(double a1, double b1, double h1, double a2, double b2, double h2, DataGridView DGV1, out double suma, out double kol)

{

double x, y, z;

x = a1;

kol = 0;

suma = 0;

int n1 = Convert.ToInt32(Math.Round((b1 - a1) / h1) + 1);

int n2 = Convert.ToInt32(Math.Round((b2 - a2) / h2) + 1);

for (int i = 1; i <= n1; i++)

{

y = a2;

for (int j = 1; j <= n2; j++)

{

z = Func(x, y);

VivodDGV(x, y, z, DGV1);

if (z > 0)

{

kol += 1;

suma += z;

}

y += h2;

}

x += h1;

}

}

public static void Zapis(DataGridView grid, TextBox t1, TextBox t2, TextBox t3, TextBox t4, TextBox t5, TextBox t6, TextBox t7)

{

SaveFileDialog sfd = new SaveFileDialog();

sfd.ShowDialog();

string filename = sfd.FileName + ".txt";

try

{

StreamWriter Z = File.CreateText(filename);

Z.WriteLine(" Исходные данные");

Z.WriteLine(" X Y");

string ax = Convert.ToString(t1.Text);

string bx = Convert.ToString(t2.Text);

string hx = Convert.ToString(t3.Text);

string ay = Convert.ToString(t4.Text);

string by = Convert.ToString(t5.Text);

string hy = Convert.ToString(t6.Text);

string y = Convert.ToString(t7.Text);

Z.WriteLine("Начальное значение:" + String.Format("{0,5}", ax) + String.Format("{0,10}", ay));

Z.WriteLine("Конечное значение: " + String.Format("{0,5}", bx) + String.Format("{0,10}", by));

Z.WriteLine("Шаг: " + String.Format("{0,5}", hx) + String.Format("{0,10}", hy));

Z.WriteLine("");

Z.WriteLine(" Таблица результатов");

Z.WriteLine(" X Y Z");

for

(int i = 0; i < grid.RowCount; i++)

{

string str1 = Convert.ToString(grid[0, i].Value);

string str2 = Convert.ToString(grid[1, i].Value);

string str3 = Convert.ToString(grid[2, i].Value);

Z.WriteLine(String.Format("{0,10}", str1) + String.Format("{0,10}", str2) + String.Format("{0,10}", str3));

}

Z.WriteLine("Наибольшее значение z:" + String.Format("{0,5}", y));

Z.Close();

MessageBox.Show("Запись успешна", "Запись", MessageBoxButtons.OK, MessageBoxIcon.Information);

Process.Start(filename);

}

catch (Exception e)

{

MessageBox.Show("Случилось что-то непоправимое " + e.Message, "Ошибка", MessageBoxButtons.OKCancel, MessageBoxIcon.Error);

}

}

public static void Prof(TextBox textBox1, KeyPressEventArgs e)

{

int x1;

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

// Если нажатая клавиша не цифровая.

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

// Запрет на ввод более одного знака минуса.

if (e.KeyChar != '-' || textBox1.Text.IndexOf('-') != -1)

// Запрет на ввод более одного десятичного разделителя.

if (e.KeyChar != ds || textBox1.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 == (char)Keys.D0 || e.KeyChar == (char)Keys.NumPad0)

if (textBox1.Text.Length >= 1) // Если строка не пустая

if (textBox1.Text[0] == '0' && textBox1.SelectionStart < 2)

e.Handled = true; // Запрет ввода

//-незначащий ноль

if (Char.IsDigit(e.KeyChar))

if (textBox1.Text.Length == 1 && textBox1.Text[0] == '0')

textBox1.Paste(",");

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

if (e.KeyChar == ds)

{ // Если введен десятичный разделитель

x1 = 0;

if (textBox1.Text != "" && textBox1.Text[0] == '-') { x1 = 1; }

if (textBox1.Text.IndexOf(ds) == -1 && textBox1.SelectionStart == x1)

{

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

textBox1.Text = textBox1.Text.Insert(x1, "0"); // Вставка 0

textBox1.SelectionStart = x1 + 1; // Текстовый курсор в конец

}

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

}

if (e.KeyChar == '-' && textBox1.Text.IndexOf('-') == -1)

textBox1.SelectionStart = 0; // Текстовый курсор в начало

//-незначащий ноль

}

}

}

**4. Результат выполнения программы**

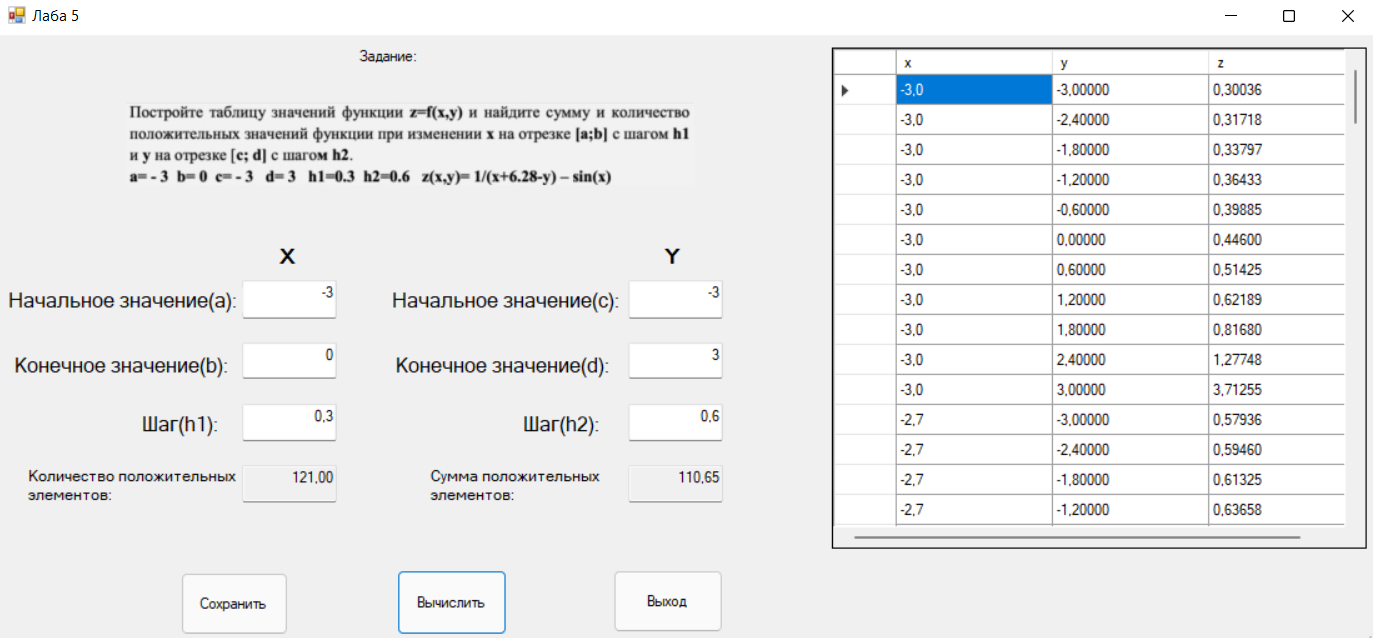


Рисунок 10 – Результат выполнения программы

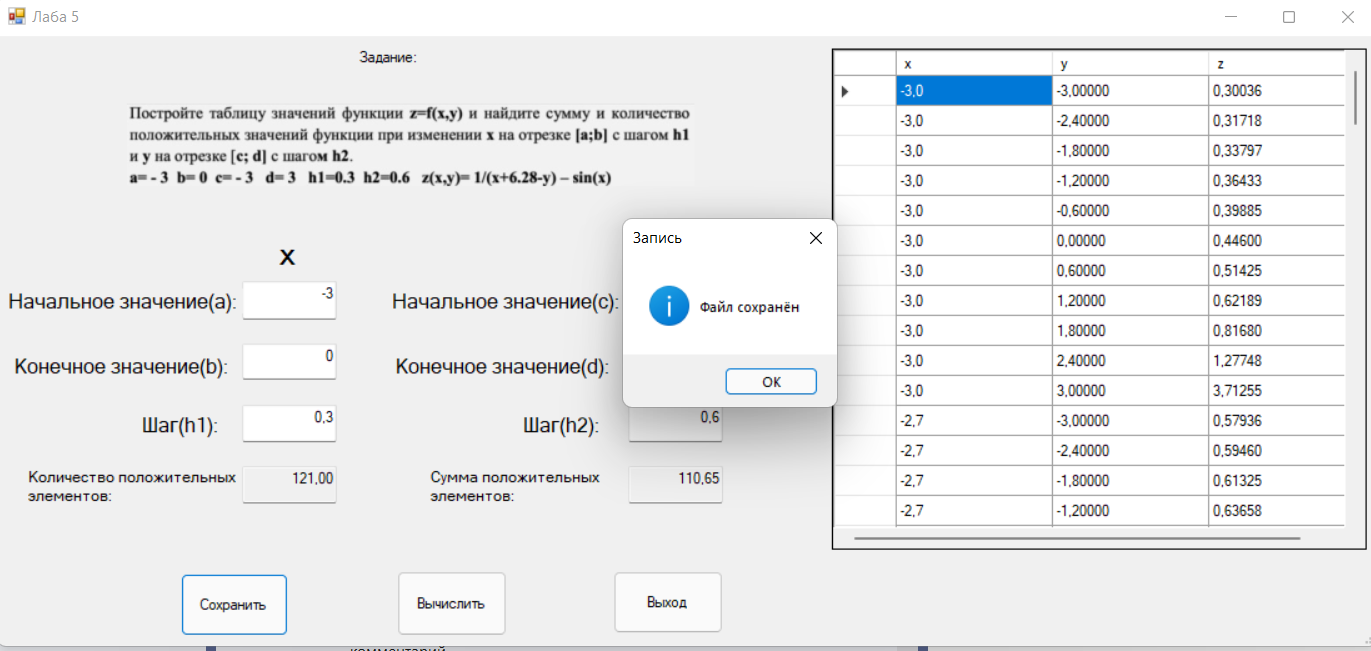


Рисунок 11 – Сохранение таблицы в файл

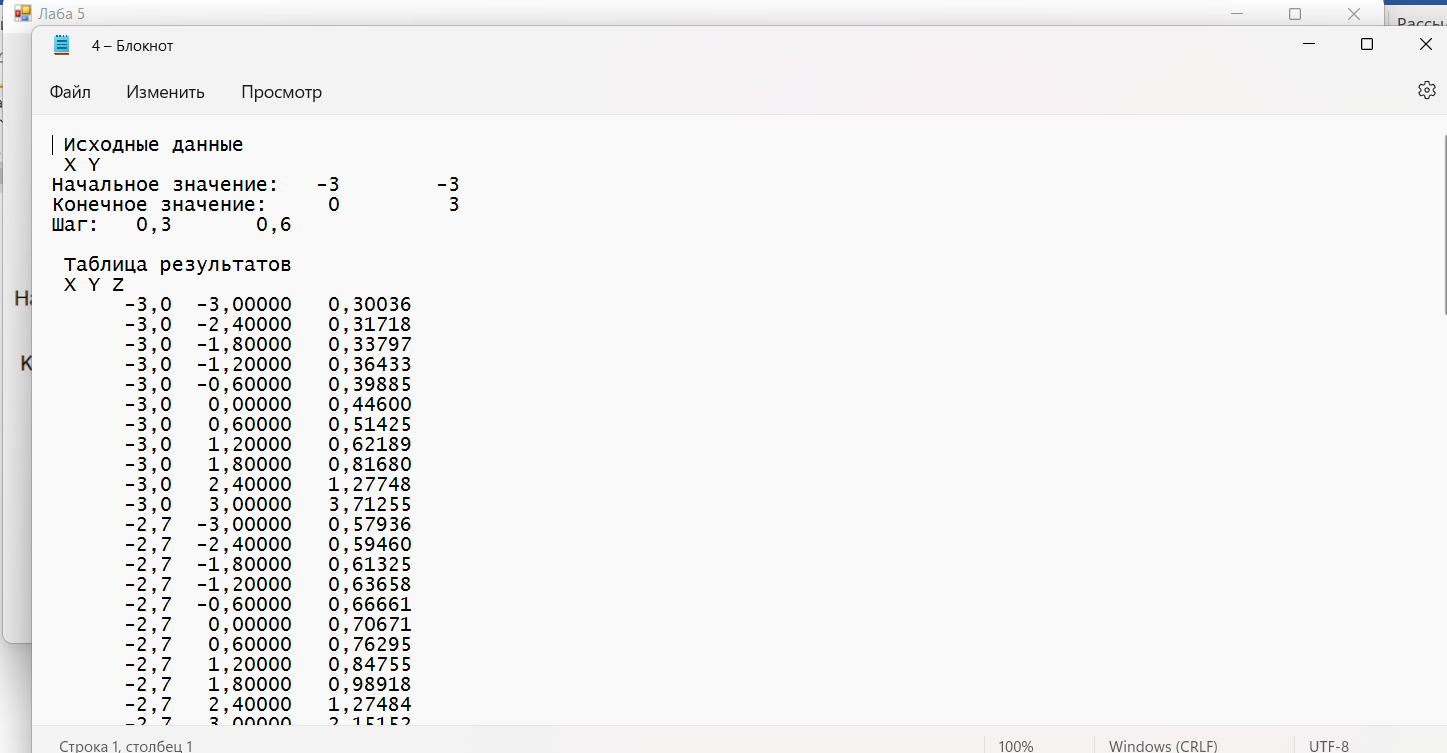


Рисунок 12 – Результат сохранение таблицы в файл

**5. Доказательство работы программы**

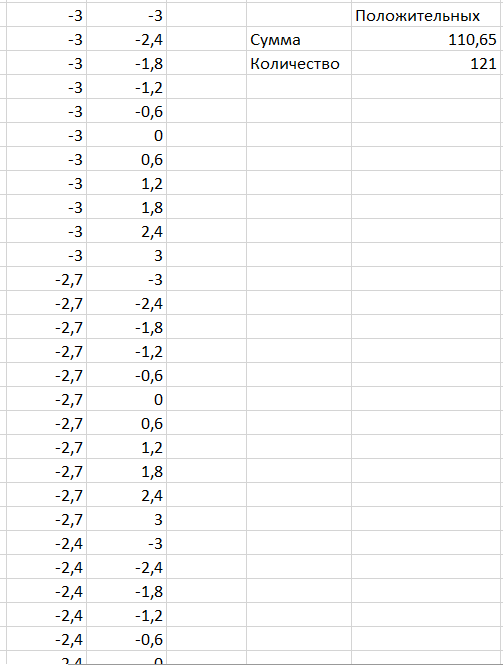


Рисунок 13 – Доказательство вычислений с помощью Microsoft Excel