**Алимов Тимур Русланович, группа 201-324** МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»**

**(МОСКОВСКИЙ ПОЛИТЕХ)**

09.03.01 Информатика и вычислительная техника

Образовательная программа (профиль)

«Интеграция и программирование в САПР»

Кафедра «СМАРТ технологии»

ОТЧЕТ

по дисциплине:

**Проектная деятельность**

на тему:

Численное интегрирование

Преподаватель: / Толстиков А.В., к.т.н. /

*подпись ФИО, уч. звание и степень*

Студент: / Сергичева В.Н., 191-324 /

*подпись ФИО, группа*

Москва, 2022 г.

## **Теоретическая часть**

Численное интегрирование - вычисление значения определённого интеграла. Под численным интегрированием понимают набор численных методов для нахождения значения определённого интеграла. Численное интегрирование применяется, когда: Сама подынтегральная функция не задана аналитически.

В данной работе мы будем применять формулы Адамса 2 порядка, Милна 2 порядка и Формулу Хемминга

**Явная формула Адамса второго порядка (Рисунок 1)**

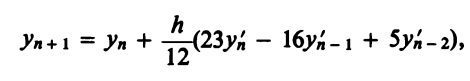


Рисунок 1 - Явная формула Адамаса второго порядка

**Неявная формула Милна первого порядка (Рисунок 2)**

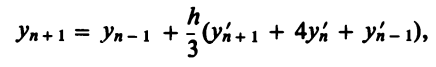


Рисунок 2 - Неявная Формула Милна первого порядка

**Формула Хемминга (Рисунок 3)**

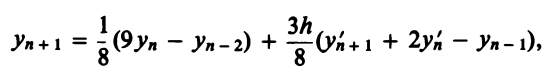
Если задача сформулирована не в канонической форме а в неявном виде мы можем ее продифференцировать и получить (Рисунок 4)

Рисунок 4 - Дифференцируем

Рисунок 3 - Формула Хемминга



Наиболее рациональным решением нашего уравнение будет решение с помощью следующих формул (Рисунок 5)

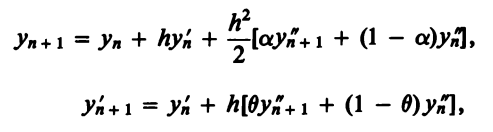
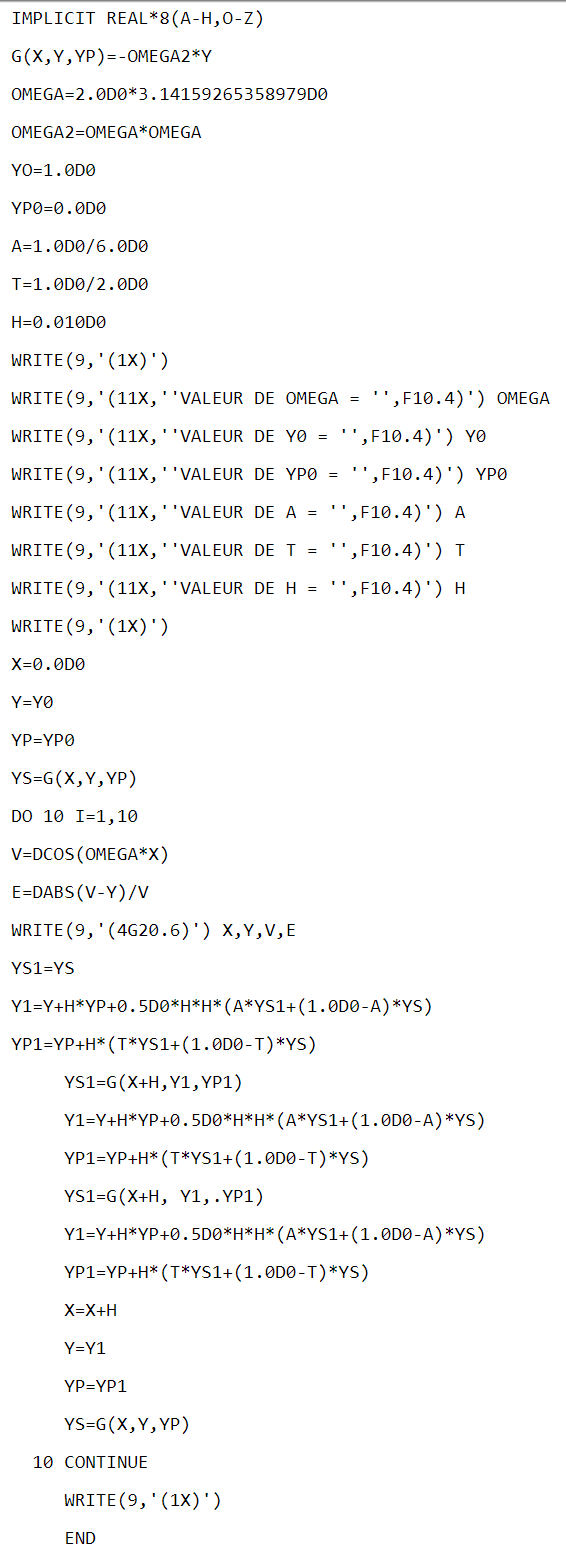
Данный метод оптимален при А=1/6, T=1/2.

Рисунок 5 - Подходящие формулы

**Практическая часть**

Для примера дан код на языке Fortran (Листинг 1).



Листинг 1 - Код на Fortran

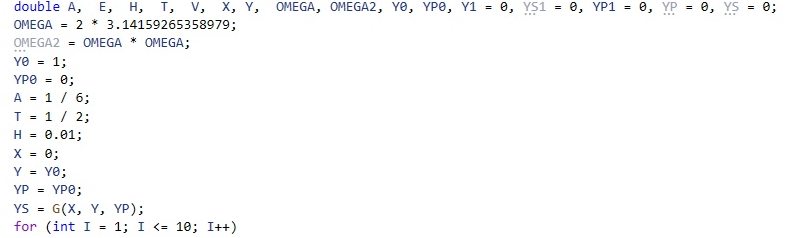
Будем переводить код на C# с помощью Windows Forms (.NET Framework).

На форме создаем структуру окна: добавляем Lable, TextBox, Button. В окне свойств добавляем необходимые события для этих элементов.

В строке 1 программы на языке Fortran написана инициализация возможных переменных.

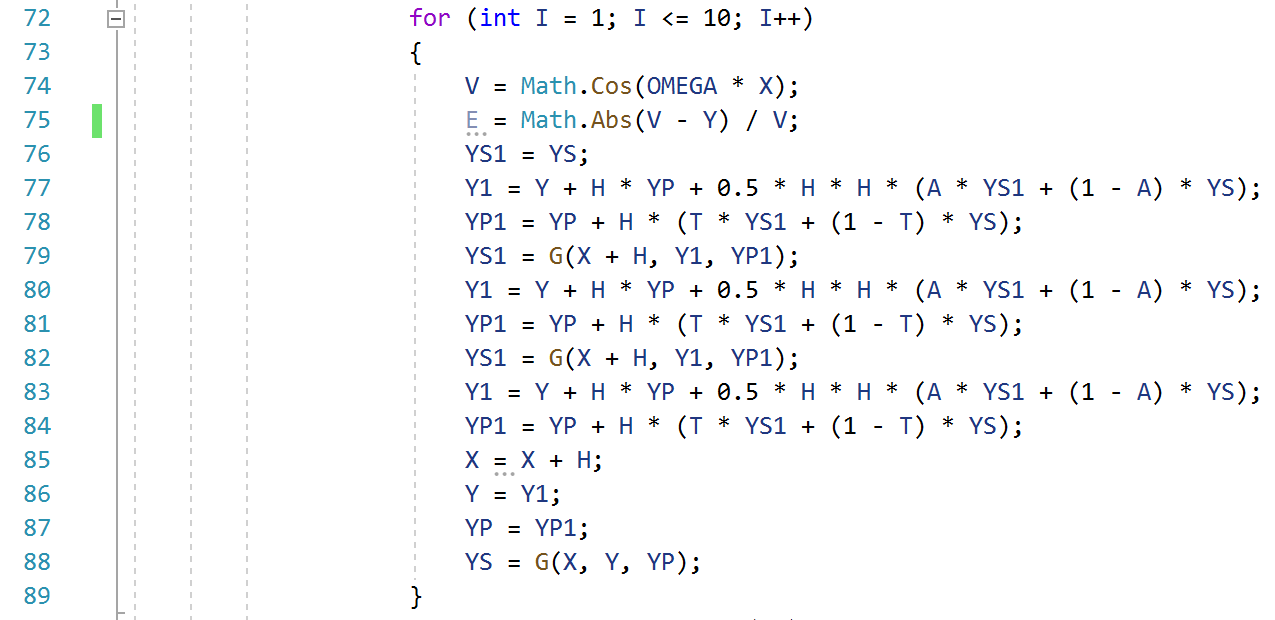
В строке 2 проинициализирована константа.

Инициализация переменных представлена на Листинге 2



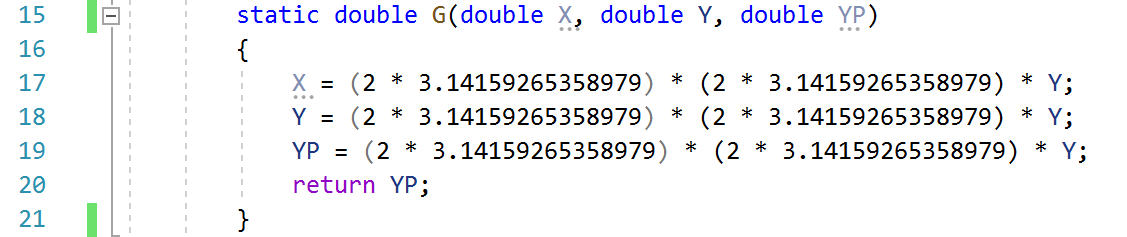
Листинг 2 - Инициализация переменных

В строке 72 начало цикла DO, который продолжается до метки 89. Реализация данного цикла на C# расположена в строках (Листинг 3).



Листинг 3 - Реализация цикла DO

В строке 2 программы на языке Fortran прописана функция G. Также она прописана в строках 15-21 на языке C# (Листинг 4)



Листинг 4 - Функция G

Расчет переменных v и e представлен в строках 40, 41 для Fortran и 74, 76 для C# (Листинг 5).

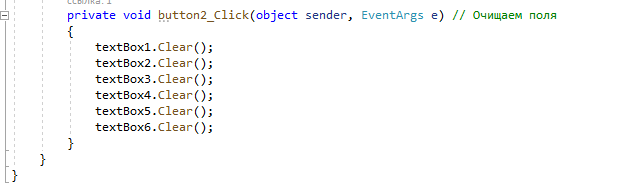


Листинг 5 - Переменные v и e

В строках 43-55 программы на языке Fortran производится расчет коэффициентов. На строках 90-91 выполняется вывод переменных (Листинг 6).



Листинг 6 -Вывод переменных

Кнопка Очистить (Листинг 7).

Листинг 7 - Кнопка очистить

Интерфейс нашей программы представлен на Рисунке 6.

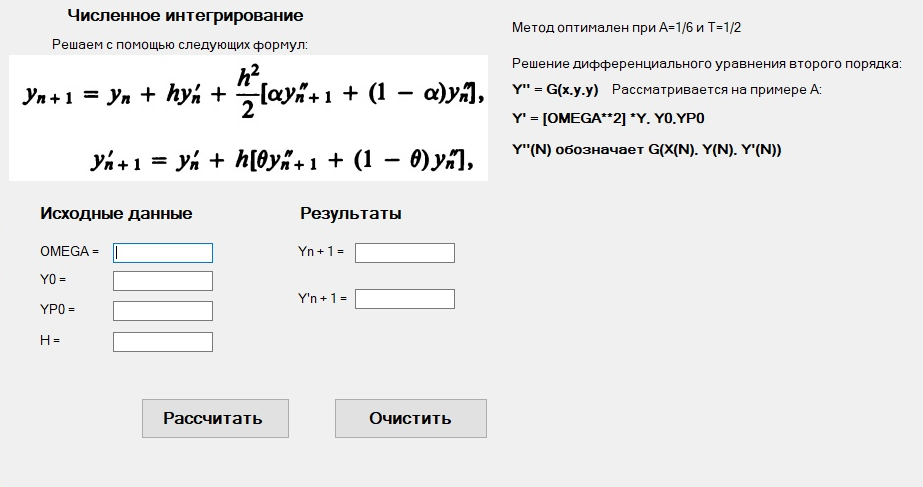


Рисунок 6 - Интерфейс программы

### **Результаты**

В результате работы сделана программа по расчету дифференциального уравнения второго порядка. С экрана считываются некоторые переменные, и также коэффициенты выводятся на экран.

ПРИЛОЖЕНИЕ 1. Код файла 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 Proect2

{

public partial class Form1 : Form

{

static double G(double X, double Y, double YP) // Функция G

{

X = (2 \* 3.14159265358979) \* (2 \* 3.14159265358979) \* Y;

Y = (2 \* 3.14159265358979) \* (2 \* 3.14159265358979) \* Y;

YP = (2 \* 3.14159265358979) \* (2 \* 3.14159265358979) \* Y;

return YP;

}

public Form1()

{

InitializeComponent();

}

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

{

}

private void textBox2\_TextChanged(object sender, EventArgs e)

{

}

private void textBox3\_TextChanged(object sender, EventArgs e)

{

}

private void textBox4\_TextChanged(object sender, EventArgs e)

{

}

private void textBox5\_TextChanged(object sender, EventArgs e)

{

}

private void textBox6\_TextChanged(object sender, EventArgs e)

{

}

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

{

double A, E, H, T, V, X, Y, OMEGA, OMEGA2, Y0, YP0, Y1 = 0, YS1 = 0, YP1 = 0, YP = 0, YS = 0; // Инициализация всех переменных

OMEGA = 2 \* 3.14159265358979;

OMEGA2 = OMEGA \* OMEGA;

Y0 = 1;

YP0 = 0;

A = 1 / 6;

T = 1 / 2;

H = 0.01;

X = 0;

Y = Y0;

YP = YP0;

YS = G(X, Y, YP);

for (int I = 1; I <= 10; I++)// Реализация цикла DO

{

V = Math.Cos(OMEGA \* X);

E = Math.Abs(V - Y) / V;

YS1 = YS;

Y1 = Y + H \* YP + 0.5 \* H \* H \* (A \* YS1 + (1 - A) \* YS);

YP1 = YP + H \* (T \* YS1 + (1 - T) \* YS);

YS1 = G(X + H, Y1, YP1);

Y1 = Y + H \* YP + 0.5 \* H \* H \* (A \* YS1 + (1 - A) \* YS);

YP1 = YP + H \* (T \* YS1 + (1 - T) \* YS);

YS1 = G(X + H, Y1, YP1);

Y1 = Y + H \* YP + 0.5 \* H \* H \* (A \* YS1 + (1 - A) \* YS);

YP1 = YP + H \* (T \* YS1 + (1 - T) \* YS);

X = X + H;

Y = Y1;

YP = YP1;

YS = G(X, Y, YP);

}

textBox5.Text = Convert.ToString(Y1); // Вывод итогов 1

textBox6.Text = Convert.ToString(YP1);// Вывод итогов 2

}

private void label4\_Click(object sender, EventArgs e)

{

}

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

{

}

private void label6\_Click(object sender, EventArgs e)

{

}

private void label7\_Click(object sender, EventArgs e)

{

}

private void label8\_Click(object sender, EventArgs e)

{

}

private void label9\_Click(object sender, EventArgs e)

{

}

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

{

textBox1.Clear();

textBox2.Clear();

textBox3.Clear();

textBox4.Clear();

textBox5.Clear();

textBox6.Clear();

}

}

}