Министерство образования и науки Российской Федерации

Рязанский государственный радиотехнический университет

Кафедра вычислительной и прикладной математики

Лабораторная работа №7

«Решение обыкновенных дифференциальных уравнений»

Выполнил студент гр. 744 Ванюков А.А.

Проверил доцент кафедры ВПМ

Проказникова Е.Н.

Рязань 2019

Цель работы: получить навыки решения обыкновенных дифференциальных уравнений, систем дифференциальных уравнений и дифференциальных уравнений второго порядка на ЭВМ при помощи численного аналитического метода Рунге-Кутта; построить графики решения уравнений данным методом для разного значения шага.

**Задание на лабораторную работу.**

**Вариант 13.** Задание 2.1: Решить на отрезке [x0, хend] задачу Коши y’ = f(x,y), y(x0) = y0 методом Рунге-Кутта с постоянным шагом. Изобразить графики решений, вычисленных с шагами h, 2h, h/2

|  |  |  |  |
| --- | --- | --- | --- |
| **Вариант** | **f(x,y,y’)=0** | **Начальное условие** | **хend** |
| **2** |  | y(1)=e | 2.6 |

Задание 2.2: Решить задачу Коши y’1 = f1(x,y1,y2), y’2 = f2(x,y1,y2), y1(a) = y1,0, y2(a) = y2,0 на отрезке [a,b] методом Рунге-Кутта с постоянным шагом h=0,1. Изобразить графики решений, вычисленных с шагом h, 2h, h/2.

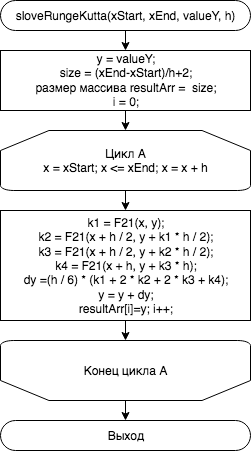
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Вариант** | **f1(x,y1,y2)** | **f2(x,y1,y2)** | **y1(a)** | **y2(a)** | **а** | **b** |
| **2** |  |  | 0.5 | 1.5 | 0 | 2 |

Задание 2.3. Найти общее решение линейного однородного дифференциального уравнения второго порядка y’’ + a1y’ + a2y = 0. Решить задачу Коши y’’ + a1y’ + a2y = 0, y(a) = y0, y’(a) = y1. Изобразить его график.

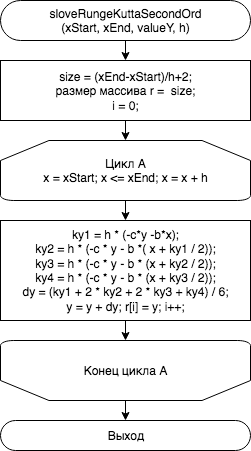
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Вариант** | **а1** | **а2** | **y(a)** | **y’(a)** | **а** |
| 2 | -4 | 4 | 0 | 1 | 0 |

**Разработка алгоритма решения**

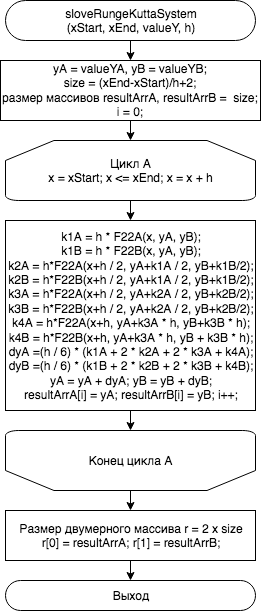
Задание 2.1



Задание 2.3



Задание 2.2



**Рзаработка программы.**

Разработка осуществлялась в среде Visual Studio C#

Функции для варианта:

﻿using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Drawing.Drawing2D;

using System.Linq;

using System.Text;

using System.Threading.Tasks;

using System.Windows.Forms;

namespace Task1

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

private Font drawFont;

private Graphics graph;

private Bitmap bmp;

private Pen pen;

private Brush drawBrush;

public Graphics Graph { get => graph; set => graph = value; }

public Bitmap Bmp { get => bmp; set => bmp = value; }

public Pen Pen { get => pen; set => pen = value; }

public Font DrawFont { get => drawFont; set => drawFont = value; }

public Brush DrawBrush { get => drawBrush; set => drawBrush = value; }

int a = -3;//интервал по Х; a и b должно нацело делится на dx:

int b = 3;

double dx = 0.5;

int fmin = -3;//интервал по Y; fmin и fmax должно нацело делится на dy:

int fmax = 3;

double dy = 0.5;

private double f(double x, double y, int p)

{

switch(p)

{

case 1:

return y \* Math.Log(y) / (-x);

case 2:

return Math.Atan(x \* x + y \* y);

case 3:

return Math.Sin(x + y);

default:

return 0;

}

}

private double[,] CalculateRungeKutte(double x, double y, double xn, double h, double kh, double[,] array, int p)

{

double h1 = h \* kh;

double k1, k2, k3, k4;

int i = 0;

int j = 0;

for (x = x; x <= xn-h1/2; x += h1)

{

k1 = h1 \* f(x, y, p);

k2 = h1 \* f(x + h1 / 2, y + k1 / 2, p);

k3 = h1 \* f(x + h1 / 2, y + k2 / 2, p);

k4 = h1 \* f(x + h1, y + k3, p);

y = y + (k1 + 2 \* k2 + 2 \* k3 + k4) / 6;

array[i, j] = x;

j = 1;

array[i, j] = y;

if (dataGridView1.Columns.Count == 3)

dataGridView1.Rows.Add(x, y, h1);

else

dataGridView1.Rows.Add(p-1, x, y, h1);

i++;

j = 0;

}

return array;

}

private void DrawFunc(/\*double x, double y,\*/ Color cl, double[,] array)

{

Pen = new Pen(cl, 3);

double mx = (pictureBox1.Width - 0) / (b - a); //масштаб по Х

double my = (pictureBox1.Height - 0) / (fmax - fmin); //масштаб по Y

for (int i = 1; i < array.GetLength(0); i++)

{

double x1 = pictureBox1.Width / 2 + array[i-1, 0] \* mx;

double y1 = pictureBox1.Height / 2 - array[i-1, 1] \* my;

double x2 = pictureBox1.Width / 2 + array[i, 0] \* mx;

double y2 = pictureBox1.Height / 2 - array[i, 1] \* my;

graph.DrawLine(pen, (float)x1, (float)y1, (float)x2, (float)y2);

}

}

private void DrawXY()

{

int x0 = pictureBox1.Width / 2;

int y0 = pictureBox1.Height / 2;

Pen = new Pen(Color.Black, 4)

{

EndCap = LineCap.ArrowAnchor//добавление "наконечника" на ось

};

graph.DrawLine(Pen, 0, y0, pictureBox1.Width, y0);//ось x

graph.DrawLine(Pen, x0, pictureBox1.Height, x0, 0);//ось y;

DrawFont = new Font("Calibri", 18);

DrawBrush = new SolidBrush(Color.Black);

graph.DrawString("x", drawFont, DrawBrush, pictureBox1.Width - 15, y0 - 30);//надпись x

graph.DrawString("y", drawFont, DrawBrush, x0 - 25, 1);//надпись y

//int a = -3;//интервал по Х; a и b должно нацело делится на dx:

//int b = 3;

//double dx = 0.5;

//int fmin = -3;//интервал по Y; fmin и fmax должно нацело делится на dy:

//int fmax = 3;

//double dy = 0.5;

double mx = (pictureBox1.Width - 0) / (b - a); //масштаб по Х

double my = (pictureBox1.Height - 0) / (fmax - fmin); //масштаб по Y

drawFont = new Font("Calibri", 12);

int n = Convert.ToInt32(((b - a) / dx) + 1);//засечки по оси OX:

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

{

double num = a + (i - 1) \* dx;//Координата на оси ОХ

int x = Convert.ToInt32(Math.Truncate(mx \* (num - a)));//Координата num в окне

graph.DrawLine(pen, x, y0 - 5, x, y0 + 5);//рисуем засечки на оси OX

if (Math.Abs(num) > 1E-15)//Исключаем 0 на оси OX

{

graph.DrawString(num.ToString(), drawFont, drawBrush, x - 8, y0 + 10);

}

}

n = Convert.ToInt32(((fmax - fmin) / dy) + 1);//засечки по оси OY:

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

{

double num = fmin + (i - 1) \* dy;//Координата на оси ОY

int y = Convert.ToInt32(Math.Truncate(my \* (num - fmin)));//Координата num в окне

graph.DrawLine(pen, x0 - 5, y, x0 + 5, y);//рисуем засечки на оси OY

if (Math.Abs(num) > 1E-15)//Исключаем 0 на оси OY

{

graph.DrawString((-num).ToString(), drawFont, drawBrush, x0 + 5, y - 7);

}

}

graph.DrawString("0", drawFont, drawBrush, x0 + 5, y0 + 4);//нулевая точка

}

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

{

Bmp = new Bitmap(pictureBox1.Width, pictureBox1.Height);

Graph = Graphics.FromImage(Bmp);

DrawXY();

pictureBox1.Image = Bmp;//вывод изображения из bmp на pictureBox

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

Bmp = new Bitmap(pictureBox1.Width, pictureBox1.Height);

Graph = Graphics.FromImage(Bmp);

DrawXY();

switch (comboBox1.SelectedIndex)

{

case 0:

label2.Visible = true;

label3.Visible = true;

label4.Visible = true;

label5.Visible = false;

label6.Visible = false;

label7.Visible = false;

dataGridView1.Rows.Clear();

label1.Text = "y'=y\*ln(y)/(-x)";

if (dataGridView1.Columns.Count >= 4)

dataGridView1.Columns.RemoveAt(0);

double x = 1;

double y = Math.E;

double xn = 2.6;

double h = 0.1;

double kh = 1;

int strings =(int) (Math.Abs(xn - x)/(kh\*h));

double[,] array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 1);

DrawFunc(Color.Blue, array);

label2.Text = "h=0.1";

label2.ForeColor = Color.Blue;

//отрисовалась линия с шагом h

kh = 2;

strings = (int)(Math.Abs(xn - x) / (kh \* h));

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 1);

DrawFunc(Color.Red, array);

label3.Text = "h=0.2";

label3.ForeColor = Color.Red;

//отрисовалась линия с шагом 2h

kh = 0.5;

strings = (int)(Math.Abs(xn - x) / (kh \* h));

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 1);

DrawFunc(Color.LightGreen, array);

label4.Text = "h=0.05";

label4.ForeColor = Color.LightGreen;

//отрисовалась линия с шагом 0.5h

break;

case 1:

label2.Visible = true;

label3.Visible = true;

label4.Visible = true;

label5.Visible = true;

label6.Visible = true;

label7.Visible = true;

dataGridView1.Rows.Clear();

if (dataGridView1.Columns.Count == 3)

{

DataGridViewColumn colFunc = new DataGridViewTextBoxColumn();

colFunc.HeaderText = "Функция";

dataGridView1.Columns.Insert(0, colFunc);

}

label1.Text = "1) y1'=arctg(x^2+y2^2); 2) y2'=in(x+y1)";

x = 0;

y = 1.5;

xn = 2;

h = 0.1;

kh = 1;

strings = (int)(Math.Abs(xn - x) / (kh \* h));

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 2);

DrawFunc(Color.Blue, array);

label2.Text = "1)h=0.1";

label2.ForeColor = Color.Blue;

//отрисовалась линия первого графика с шагом h

y = 0.5;

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 3);

DrawFunc(Color.Orange, array);

label5.Text = "2)h=0.1";

label5.ForeColor = Color.Orange;

//отрисовалась линия второго графика с шагом h

kh = 2;

y = 1.5;

strings = (int)(Math.Abs(xn - x) / (kh \* h));

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 2);

DrawFunc(Color.Red, array);

label3.Text = "1)h=0.2";

label3.ForeColor = Color.Red;

//отрисовалась линия первого графика с шагом 2h

y = 0.5;

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 3);

DrawFunc(Color.DarkSlateBlue, array);

label6.Text = "2)h=0.2";

label6.ForeColor = Color.DarkSlateBlue;

//отрисовалась линия второго графика с шагом 2h

kh = 0.5;

y = 1.5;

strings = (int)(Math.Abs(xn - x) / (kh \* h));

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 2);

DrawFunc(Color.LightGreen, array);

label4.Text = "2)h=0.05";

label4.ForeColor = Color.LightGreen; ;

//отрисовалась линия первого графика с шагом 0.5h

y = 0.5;

array = new double[strings, 2];

CalculateRungeKutte(x, y, xn, h, kh, array, 3);

DrawFunc(Color.HotPink, array);

label7.Text = "2)h=0.05";

label7.ForeColor = Color.HotPink;

//отрисовалась линия второго графика с шагом 2h

break;

case 2:

label2 .Visible = false;

label3.Visible = false;

label4.Visible = false;

label5.Visible = false;

label6.Visible = false;

label7.Visible = false;

dataGridView1.Rows.Clear();

label1.Text = "y''+a1\*y'+a2\*y=0";

if (dataGridView1.Columns.Count >= 4)

dataGridView1.Columns.RemoveAt(0);

double a = 0, b = -4, c = 4;

double x1 = -3, xEnd = 3;

double ya = 0, ya1 = 1, ky1, ky2, ky3, ky4, dy;

x = x1;

y = ya1;

h = 0.05;

strings = (int)(Math.Abs(xEnd - x) / h);

array = new double[strings+1, 2];

int j = 0;

for (int i = 0; x < xEnd; i++)

{

ky1 = h \* (-c \* y - b \* x);

ky2 = h \* (-c \* y - b \* (x + ky1 / 2));

ky3 = h \* (-c \* y - b \* (x + ky2 / 2));

ky4 = h \* (-c \* y - b \* (x + ky3 / 2));

dy = (ky1 + 2 \* ky2 + 2 \* ky3 + ky4) / 6;

y += dy;

array[i, 0] = x;

array[i, 1] = y;

if (dataGridView1.Columns.Count == 3)

dataGridView1.Rows.Add(x, y, h);

x += h;

}

DrawFunc(Color.Blue, array);

break;

}

pictureBox1.Image = Bmp;//вывод изображения из bmp на pictureBox

}

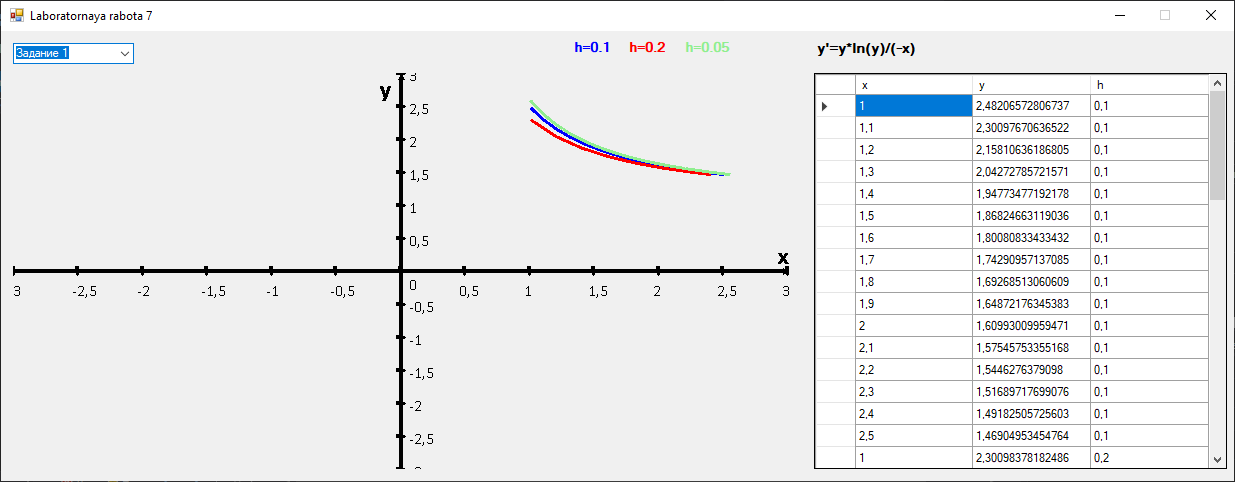
}

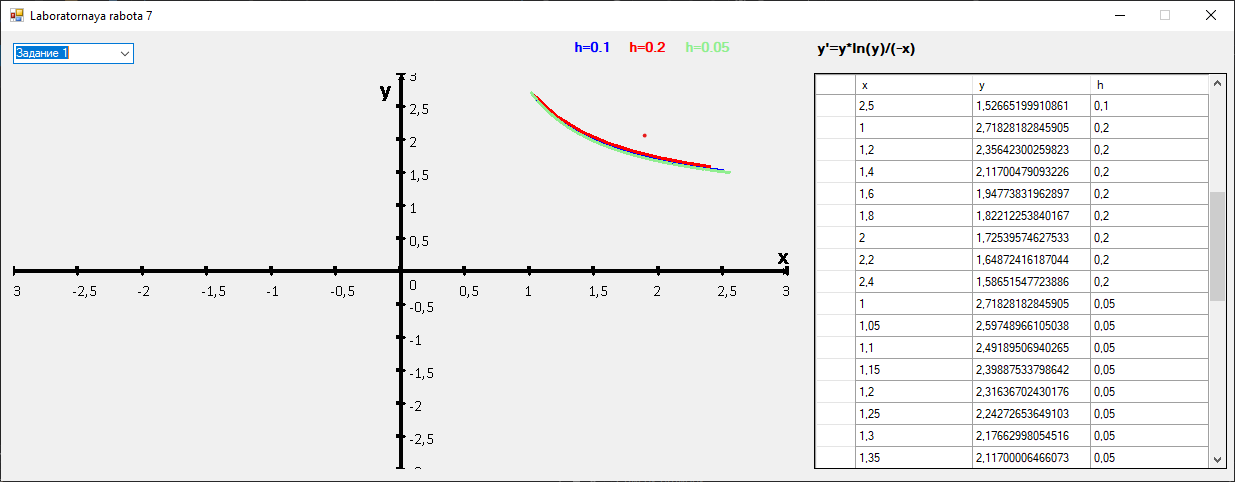
}

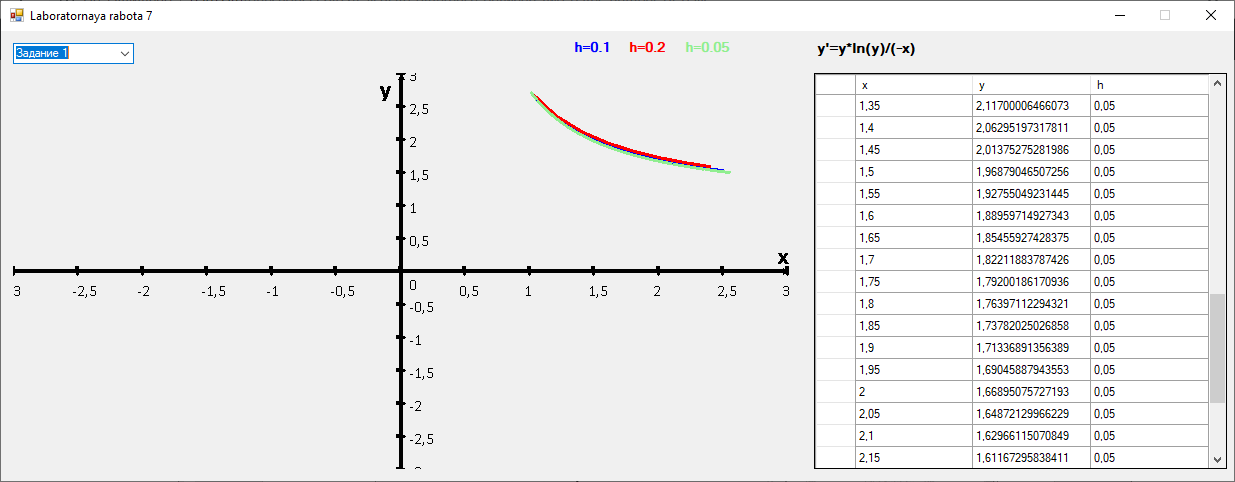
**Результат выполнения**

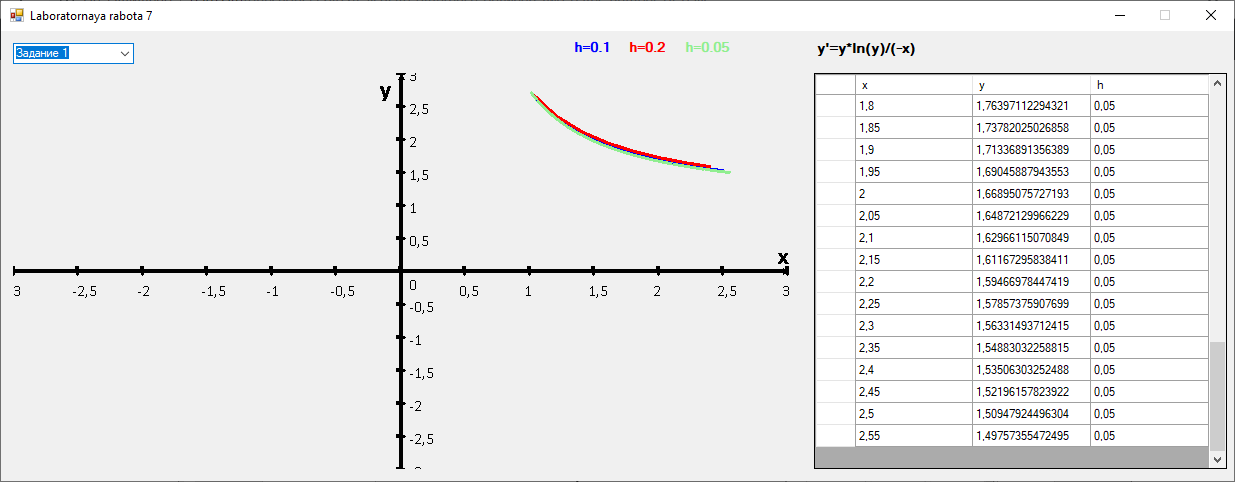
В результате рабочих прогонов программы имеем следующие результаты:

Задание 2.1:

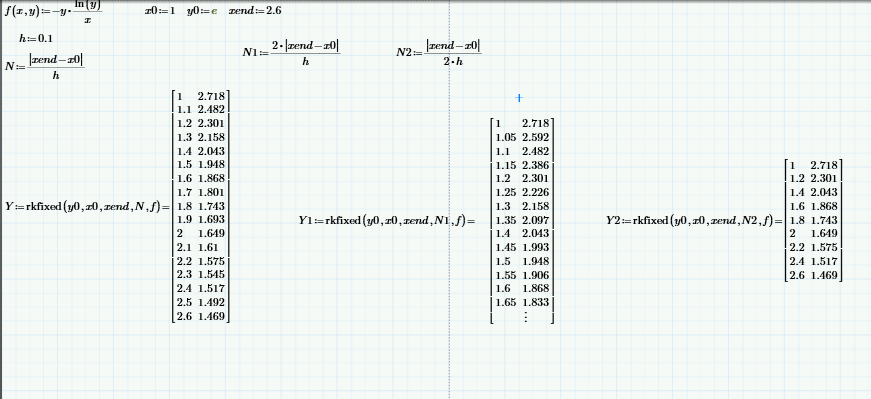


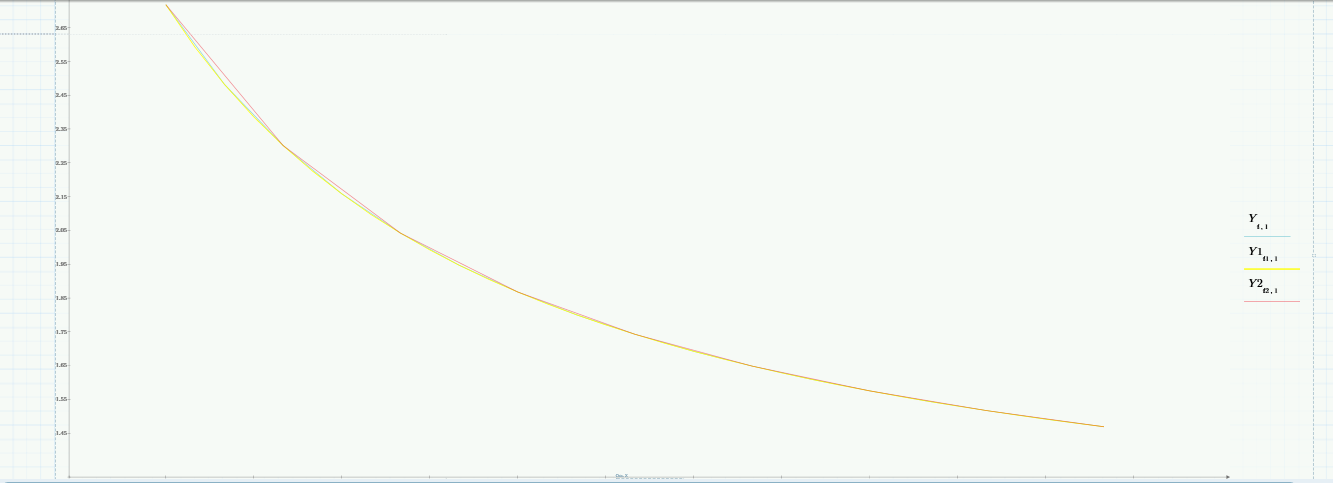




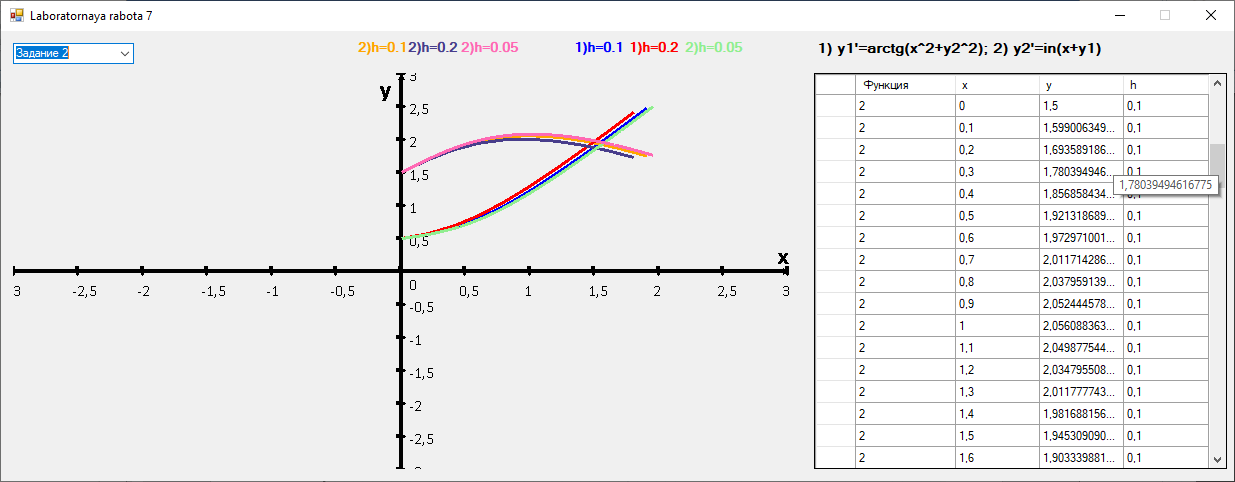
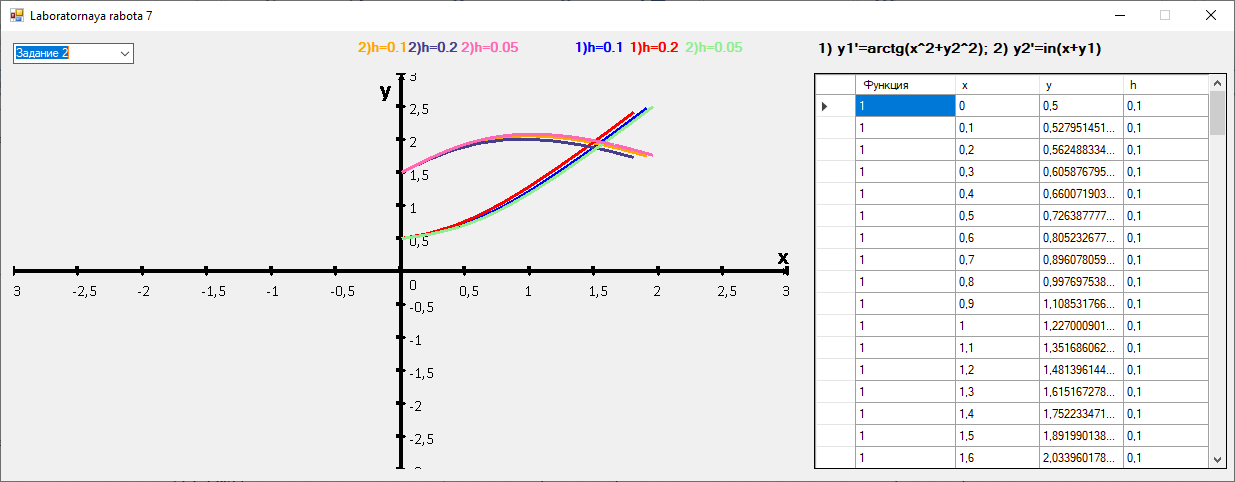


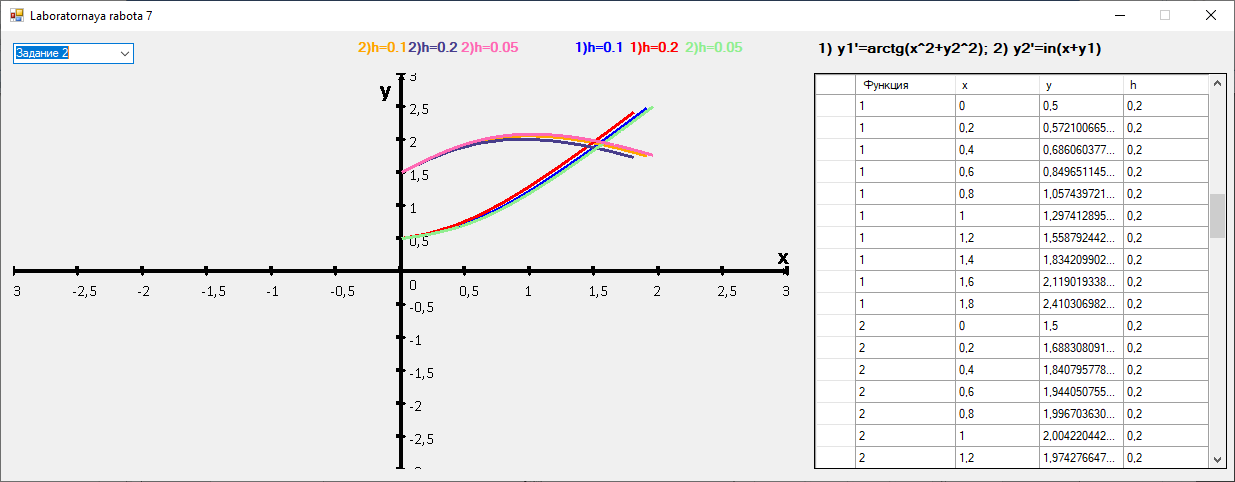
Проверка:

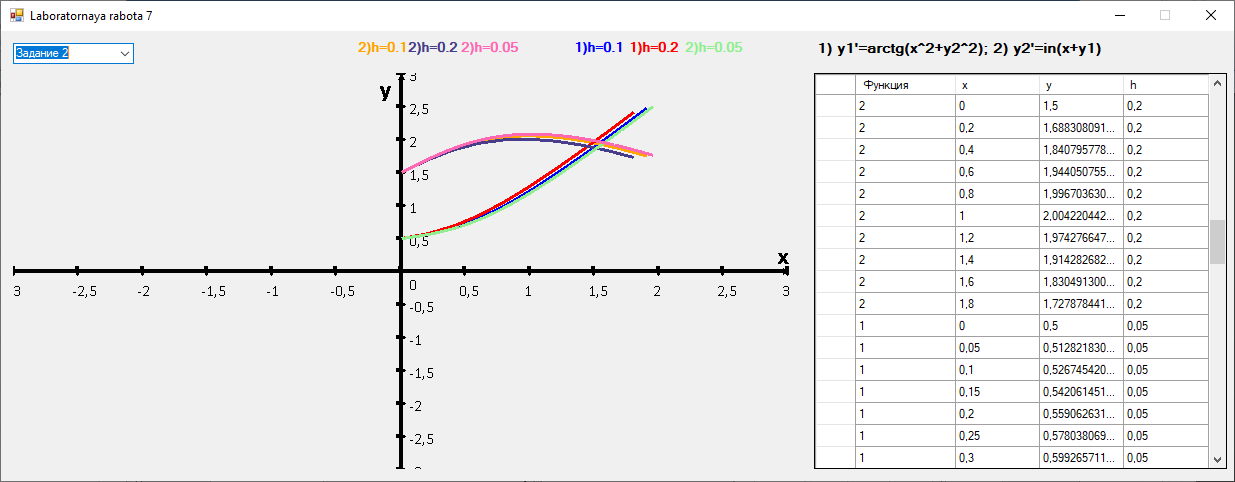


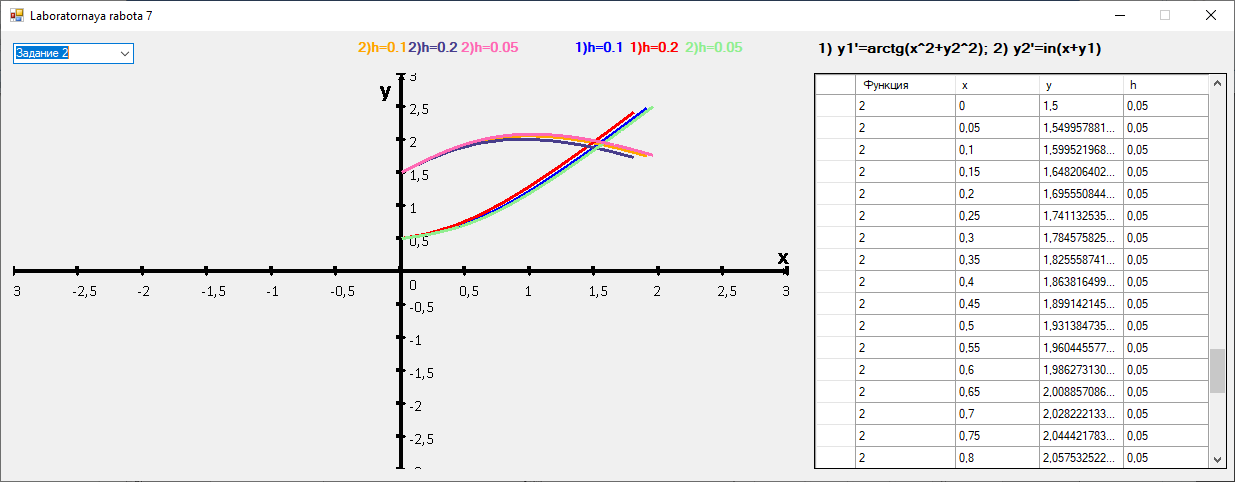


Задание 2.2:

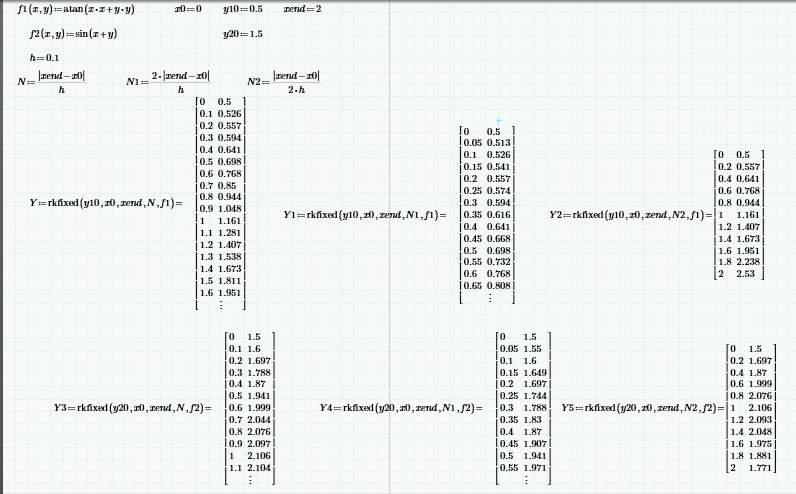


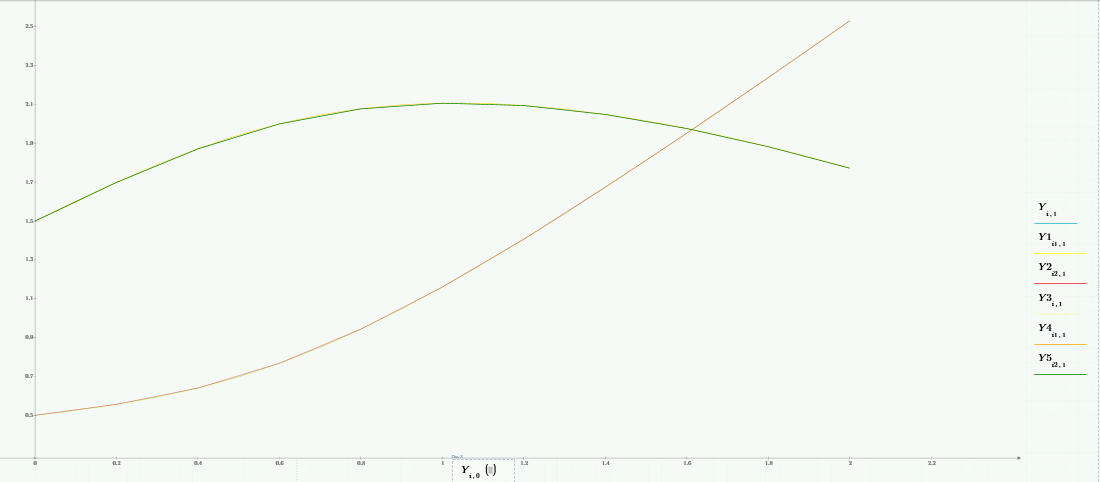




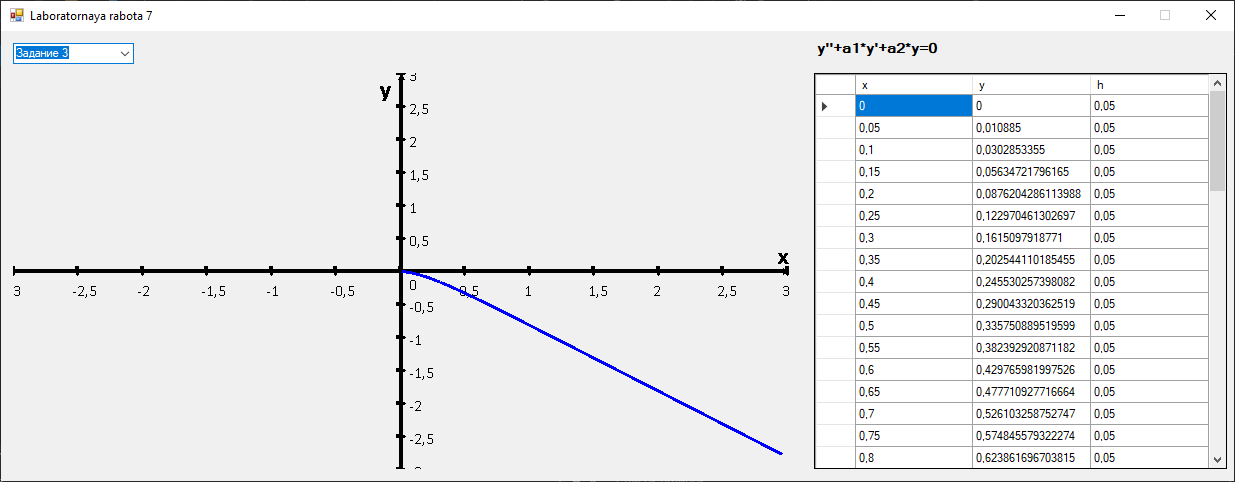


Проверка:





Задание 2.3:



Проверка:

