МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ БЕЛАРУСЬ

БЕЛОРУССКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ

***Факультет информационных технологий и робототехники***

Кафедра программного обеспечения информационных систем и технологий

**Отчет по лабораторной работе № 1,2**

по дисциплине: ”Разработка приложений в визуальных средах ”

# **на тему: *”* Создание приложения поддерживающего технологию ООП*”***

Вариант 1

Выполнил**:** студент группы 10701322 Бородкин Д.В.

Принял**:** доц. Гурский Н. Н.

Минск 2024

**Лабораторная работа № 1,2.**

**Цель лабораторной работы:**изучить принципы построения классов.

**Задание:** разработать класс вычисления определенного интеграла четырьмя методами (методами левых, правых, средних прямоугольников и трапеций). В классе предусмотреть методы графического отображения подынтегральной функции на расширенном интервале и геометрического смыслаинтеграла.

**Текст программы:**

using System.Windows.Forms;

using System;

using System.Windows.Forms.DataVisualization.Charting;

namespace Lab\_1

{

public partial class Form1 : Form

{

Integral I;

public Form1()

{

InitializeComponent();

textBox1.Text = "1";

textBox2.Text = "2";

textBox3.Text = "0,03";

I = new Integral();

}

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

{

I.a = Convert.ToDouble(textBox1.Text);

I.b = Convert.ToDouble(textBox2.Text);

I.h = Convert.ToDouble(textBox3.Text);

I.chart = chart2;

// Выбор метода в зависимости от выбранного радиобаттона

double result = 0.0;

if (radioButton1.Checked)

{

result = I.rectanglleft();

I.drawLeft();

}

else if (radioButton2.Checked)

{

result = I.rectanglright();

I.drawRight();

}

else if (radioButton3.Checked)

{

result = I.rectanglmiddle();

I.drawMiddle();

}

else if (radioButton4.Checked)

{

result = I.trap();

I.drawTrap();

}

// Отображаем результат в textBox4

textBox4.Text = result.ToString("F5");

}

}

}

Отдельный класс:

using System;

using System.Collections.Generic;

using System.Drawing;

using System.Windows.Forms.DataVisualization.Charting;

namespace Integral

{

internal class OprIntegral

{

//Поля

public double a; // нижняя граница

public double b; // верхняя граница

public double h; // шаг

public double rez; // результат

public Chart chart;

public List<double> X, Y;

//Метод для функции

public double f(double x)

{

return Math.Sin(x) / Math.Sqrt(x);

}

//Метод для трапеций

public double Trap()

{

X = new List<double>();

Y = new List<double>();

double sum = (f(a) + f(b)) / 2.0;

X.Add(a);

Y.Add(f(a));

for (double x = a + h; x < b; x += h)

{

X.Add(x);

Y.Add(f(x));

sum += f(x);

}

rez = sum \* h;

return rez;

}

//Метод для левых прямоугольников

public double Left()

{

X = new List<double>();

Y = new List<double>();

double sum = 0.0;

for (double x = a; x < b; x += h)

{

sum += f(x);

X.Add(x);

Y.Add(f(x));

}

rez = sum \* h;

return rez;

}

//Метод для правых прямоугольников

public double Right()

{

X = new List<double>();

Y = new List<double>();

double sum = 0.0;

for (double x = a + h; x <= b; x += h)

{

sum += f(x);

X.Add(x);

Y.Add(f(x));

}

rez = sum \* h;

return rez;

}

//Метод для средних прямоугольников

public double Middle()

{

double sum = 0.0;

X = new List<double>();

Y = new List<double>();

for (double x = a + h / 2; x <= b; x += h)

{

sum += f(x);

X.Add(x);

Y.Add(f(x));

}

rez = sum \* h;

return rez;

}

// Метод для рисования функции

public void drawF(SeriesChartType tp)

{

// Очищаем существующие данные на графике

chart.Series.Clear();

// Создание нового объекта класса Series для графика линии

Series series = new Series

{

Name = "График",

ChartType = SeriesChartType.Spline

};

// Установка свойств графика (например, подписей осей)

chart.ChartAreas[0].AxisX.Title = "Ось X";

chart.ChartAreas[0].AxisY.Title = "Ось Y";

// Расчет значения функции для каждой точки

for (double x = a; x <= b; x += h)

{

double y = Math.Sin(x) / Math.Sqrt(x);

series.Points.AddXY(x, y);

}

// Добавление созданного объекта Series на график

chart.Series.Add(series);

chart.Series["График"].Color = Color.Red;

chart.Series["График"].BorderWidth = 3;

// Легенда

chart.Legends["Legend1"].Title = "Легенда";

chart.Legends["Legend1"].Docking = Docking.Right;

// Заголовок

chart.Titles["Title1"].Text = "Графическое изображение функции";

// Создание нового объекта класса Series для графика площади

Series series2 = new Series

{

Name = "Площадь",

ChartType = tp

};

// Заполнение значения функции для каждой точки, которые были рассчитаны при интегрировании

for (int i=0;i<X.Count;i++)

{

series2.Points.AddXY(X[i], Y[i]);

}

// Добавление созданного объекта Series на график

chart.Series.Add(series2);

chart.Series["Площадь"].Color = Color.Blue;

chart.Series["Площадь"].BackHatchStyle = ChartHatchStyle.ForwardDiagonal; // Косая штриховка

chart.Series["Площадь"].BorderWidth = 0; // Убрать границы

// Установка ширины столбцов

chart.Series["Площадь"]["PointWidth"] = "1.1";

}

// Метод для рисования левых прямоугольников

public void drawLeft()

{

drawF(SeriesChartType.Column);

}

// Метод для рисования правых прямоугольников

public void drawRight()

{

drawF(SeriesChartType.Column);

}

// Метод для рисования средних прямоугольников

public void drawMiddle()

{

drawF(SeriesChartType.Column);

}

// Метод для рисования трапецией

public void drawTrap()

{

drawF(SeriesChartType.Area);

}

}

**Результаты выполнения программы:**

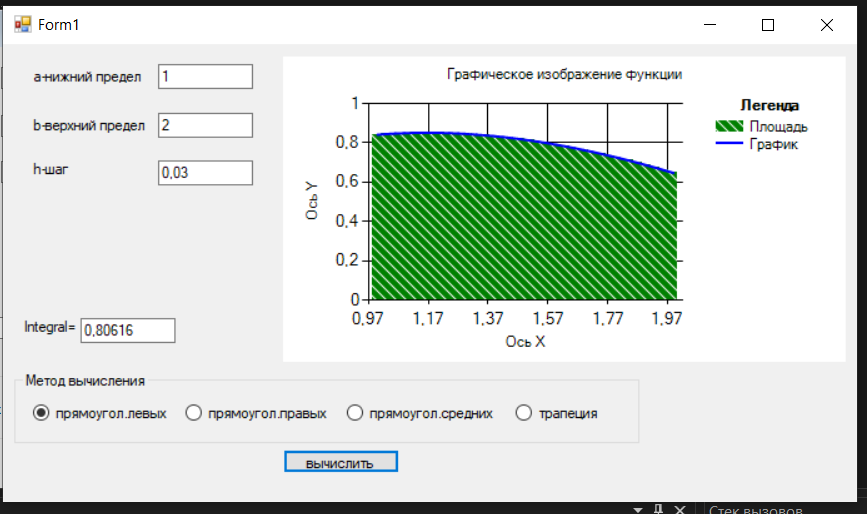


Рисунок 1. Результат работы программы.

**Вывод:** изучил принципы построения классов.