

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное  
учреждение высшего образования  
«Алтайский государственный технический университет им. И. И. Ползунова»

Факультет информационных технологий

Кафедра прикладной математики

Отчет защищен с оценкой \_\_\_\_\_  
Преподаватель Проскурин А. В.  
«\_\_» \_\_\_\_\_ 2022 г.

Отчет  
Лабораторная работа №7  
Интегрирование методом Монте-Карло  
по дисциплине «Вычислительные алгоритмы»

Студент группы ПИ 92 Д. А. Савиков, Ю. К. Гранкин

## Задание

1. Вычислите определенный интеграл функции  $f(x)$  на отрезке  $[a, b]$  методом Симпсона и Монте-Карло.
2. Отобразите графически точки испытаний методом Монте-Карло.
3. Исследуйте зависимость точности от количества испытаний.
4. Задайте на плоскости произвольную фигуру. Найдите ее центр тяжести методом Монте-Карло, предполагая плотность равномерной.

## Алгоритм

### Метод Симпсона

Формула Симпсона:

$$\int_{x_{i-1}}^{x_i} f(x) dx \approx \int_{x_{i-1}}^{x_i} L_{2,i}(x) dx = \frac{h}{6} (f(x_{i-1}) + 4f(x_{i-1/2}) + f(x_i)).$$

$$\int_a^b f(x) dx \approx \sum_{i=1}^n \int_{x_{i-1}}^{x_i} f(x) dx$$

### Метод Монте-Карло

$$\int_a^b f(x) dx \approx \frac{b-a}{N} \sum_{i=1}^N f(u_i)$$

$N$  – количество испытаний,  $u$  – очередное случайное число принадлежащее отрезку  $[a, b]$

Поиск центра тяжести прямоугольного треугольника

Координаты центра тяжести находятся по таким формулам:

$$x_0 = \frac{\iint_D x dx dy}{\iint_D dx dy}, \quad y_0 = \frac{\iint_D y dx dy}{\iint_D dx dy}$$

Кратные интегралы находятся по формуле:

$$I \approx J \frac{1}{n} \sum_{i=1}^n F(\eta_i).$$

$n$  – количество испытаний,  $\eta_i$  – случайная точка принадлежащая параллелепипеду содержащему треугольник.

$$\sum_{i=1}^n F(\eta_i).$$

В эту сумму входят только те  $\eta_i$  которые принадлежат области треугольника.

Так как во всех четырех двойных интегралах  $J$  и  $1/n$  одинаковые, то их можно сразу сократить.

В итоге получается, что  $x_0$  равно отношению суммы координаты  $x$  всех случайных точек, которые принадлежат треугольнику и количества таких точек, а  $y_0$  – также, только в числителе сумма координаты  $y$  всех случайных точек, принадлежащих треугольнику.

Текст программы

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 System.IO;

namespace AlgLab7
{
    public partial class Form1 : Form
    {
        public Form1()
        {
            InitializeComponent();
        }
        //Поиск f(x)
        public double Funk(string f, double x)
        {
            switch (f) //В зависимости от f выбирается функция
            {
                case "f(x)=x^2":
                    return x * x;
                case "f(x)=|x|":
                    return Math.Abs(x);
                case "f(x)=cos(x)":
                    return Math.Cos(x);
                case "f(x)=e^(-x*x)":
                    return Math.Exp(-x * x);
            }
            return 0;
        }
        //Поиск интеграла методом Симпсона с постоянным шагом
        //f - подинтегральная функция, N - количество сегментов на отрезке [a,b], h -
шаг
        public double Simp(double a, double b, string f, double h, int N)
        {
            double res, x;
            res = 0;
            x = a;
```

```

        for(int i = 1; i <= N; i++)
        {
            x += h;
            res += Funk(f, x - h) + 4 * Funk(f, x - h / 2) + Funk(f, x);
        }
        return res * h / 6;
    }
    //Поиск интеграла методом Монте-Карло
    //f - подинтегральная функция, N - количество испытаний
    public double Monte(double a, double b, string f, int N)
    {
        Random rand = new Random();
        double y, x, sum;
        chart1.Series[0].Points.Clear();
        chart1.ChartAreas[0].AxisX.Minimum = a;
        chart1.ChartAreas[0].AxisX.Maximum = b;
        sum = 0;
        for (int i = 0; i < N; i++)
        {
            x = rand.NextDouble() * (b - a) + a;
            y = Funk(f, x);
            sum += y;
            chart1.Series[0].Points.AddXY(x, y);
        }
        sum = sum * (b - a) / N;
        return sum;
    }
    //Кнопка поиска интеграла
    private void button1_Click(object sender, EventArgs e)
    {
        string file, funk;
        double a, b, sim, mon;
        int nMonk, nSimp;
        file = textBox1.Text;
        if (File.Exists(file))
        {
            try          //Считывание информации из файла
            {
                StreamReader stream = new StreamReader(file);
                funk = stream.ReadLine();    //строка в которой записана функция
                stream.ReadLine();
                a = Convert.ToDouble(stream.ReadLine()); //Границы отрезка
                b = Convert.ToDouble(stream.ReadLine());
                stream.ReadLine();
                nMonk = Convert.ToInt32(stream.ReadLine()); //Кол-во испытаний
                stream.ReadLine();
                nSimp = Convert.ToInt32(stream.ReadLine()); //Кол-во сегментов
                stream.Close();
            }
            catch
            {

```

интегрирования

для метода Монте-Карло

на которые надо поделить [a,b] для метода Симпсона

```

        MessageBox.Show("Данные введены не правильно");
        return;
    }
}
else
{
    MessageBox.Show("Файл не найден");
    return;
}
sim = Simp(a, b, funk, (b - a) / nSimp, nSimp);
mon = Monte(a, b, funk, nMonk);
StreamWriter str = File.AppendText(file);
str.WriteLine(); //Результаты записываются в тот же файл
str.WriteLine("Симпсон: " + sim);
str.WriteLine("Монте-Карло: " + mon);
str.Close();
}
//Поиск центра тяжести
private void button2_Click(object sender, EventArgs e)
{
    double ax, bx, ay, by, x, y, upx, upy, dow;
    ax = 1; //Границы для x и y
    ay = 1;
    bx = 6;
    by = 4;
    chart1.Series[1].Points.Clear();
    chart1.ChartAreas[0].AxisX.Minimum = 0;
    chart1.ChartAreas[0].AxisX.Maximum = bx + 1;
    chart1.ChartAreas[0].AxisX.MajorGrid.Interval = 1;
    chart1.ChartAreas[0].AxisY.Minimum = 0;
    chart1.ChartAreas[0].AxisY.Maximum = by + 1;
    chart1.ChartAreas[0].AxisY.MajorGrid.Interval = 1;
    chart1.Series[1].Points.AddXY(1, 1); //Построение прямоугольного
треугольника по координатам (1,1), (6,4), (6,1)
    chart1.Series[1].Points.AddXY(6, 4);
    chart1.Series[1].Points.AddXY(6, 1);
    chart1.Series[1].Points.AddXY(1, 1);
    Random rand = new Random();
    upx = upy = dow = 0; //upx - числитель в формуле поиска x центра
тяжести, upy - тоже самое но для y, dow - знаменатель для обеих формул
    for (int i = 0; i < 900000; i++) //900000 испытаний
    {
        y = rand.NextDouble() * (by - ay) + ay;
        x = rand.NextDouble() * (bx - ax) + ax;
        if (x >= (y - 0.4) / 0.6) //Проверка того входит ли очередная точка
в область треугольника
        {
            upy += y;
            upx += x;
            dow++;
        }
    }
    x = upx / dow;
    y = upy / dow;

```

```

        chart1.Series[0].Points.Clear();
        chart1.Series[0].Points.AddXY(x, y);    //Построение точки центра тяжести
и вывод результата
        MessageBox.Show("(" + x + " ; " + y + ")");
    }
}
}

```

## Тест программы

1)

\*1.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=x^2$

Границы

-1

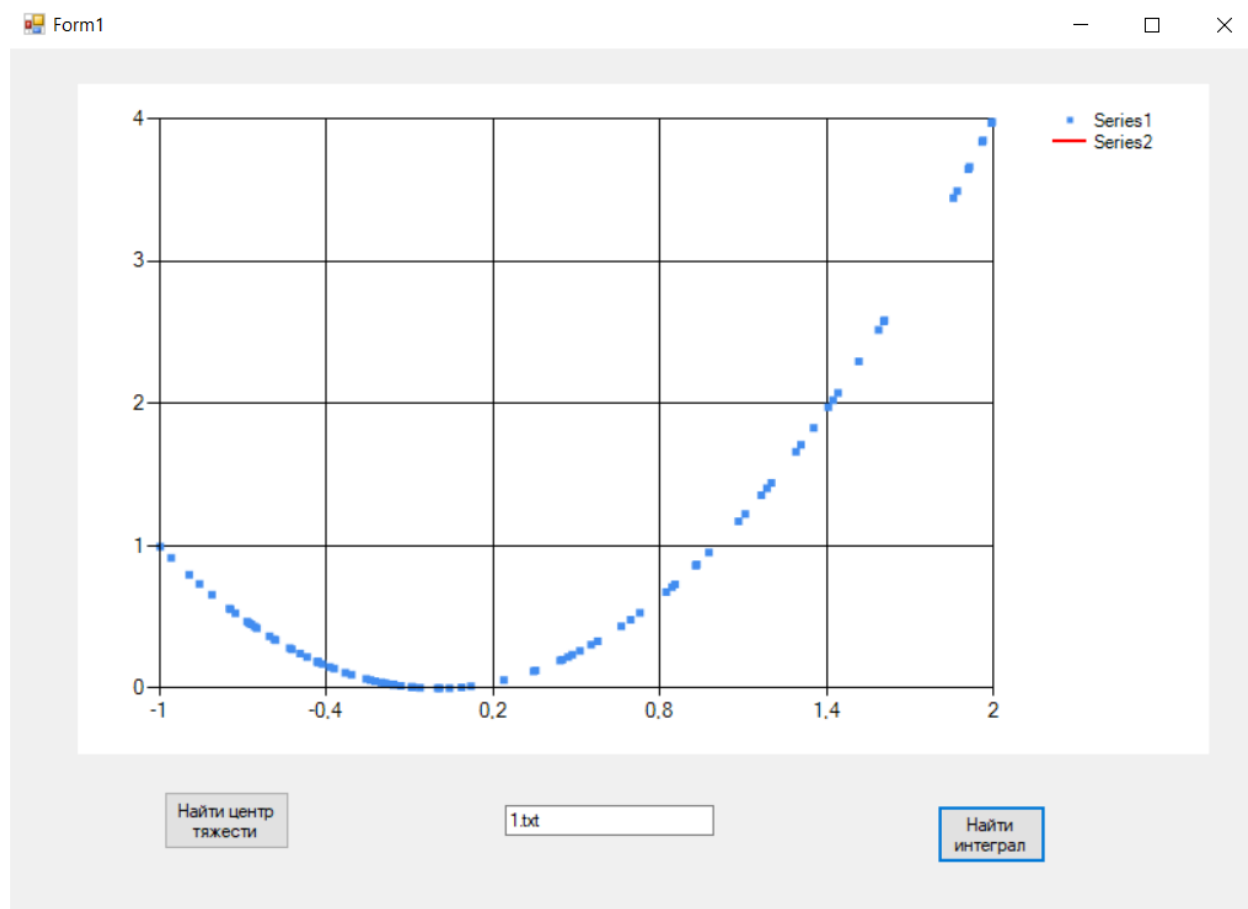
2

Кол-во испытаний для метода Монте-Карло

90

Кол-во сегментов отрезка для метода Симпсона

10



1.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=x^2$

Границы

-1

2

Кол-во испытаний для метода Монте-Карло

90

Кол-во сегментов отрезка для метода Симпсона

10

Симпсон: 3

Монте-Карло: 2,62031666735515

Увеличим кол-во испытаний

\*1.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=x^2$

Границы

-1

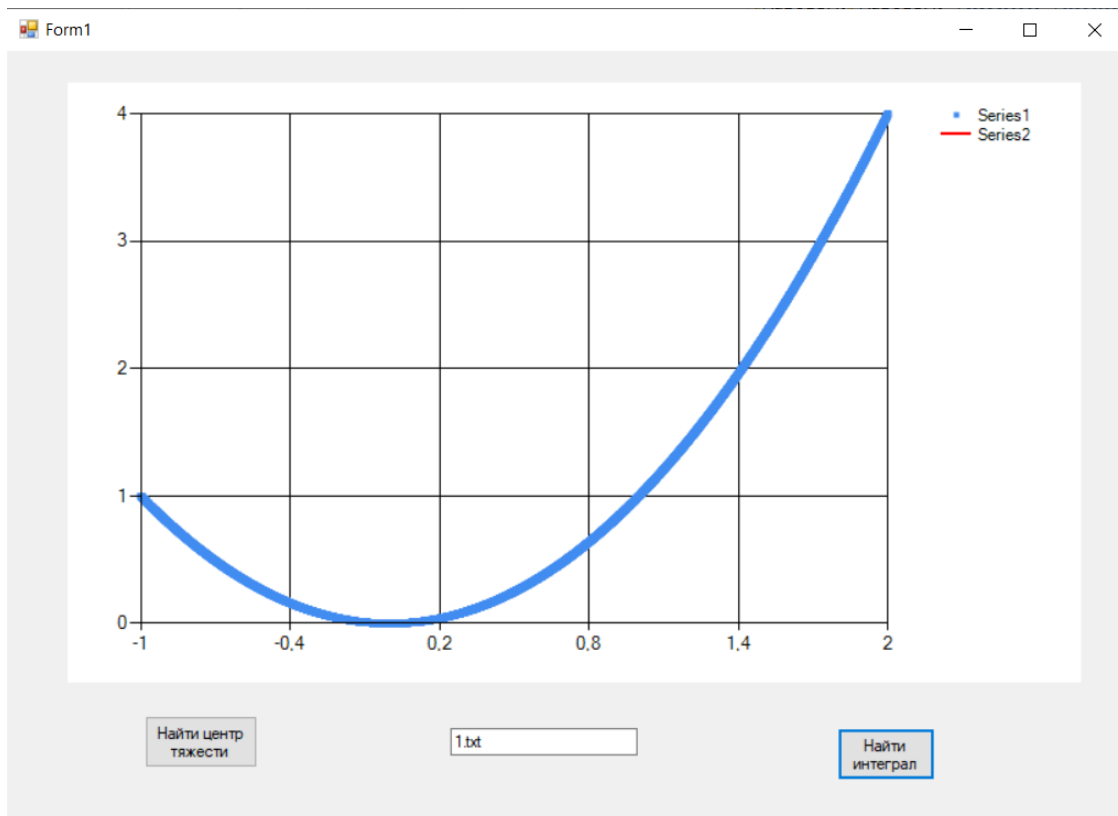
2

Кол-во испытаний для метода Монте-Карло

9000

Кол-во сегментов отрезка для метода Симпсона

10



1.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=x^2$

Границы

-1

2

Кол-во испытаний для метода Монте-Карло

9000

Кол-во сегментов отрезка для метода Симпсона

10

Симпсон: 3

Монте-Карло: 3,01327026298635

Еще раз увеличим кол-во испытаний

\*1.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=x^2$

Границы

-1

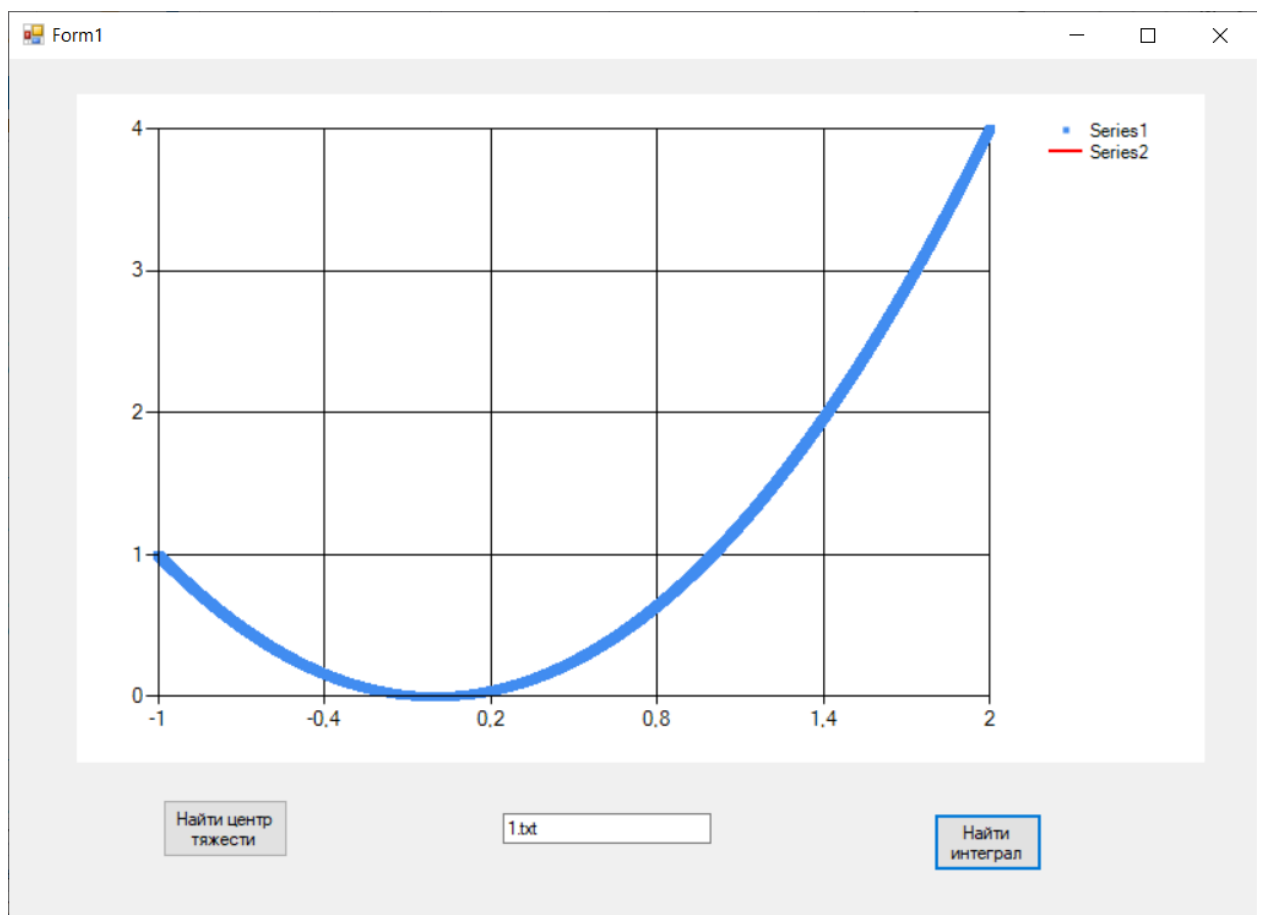
2

Кол-во испытаний для метода Монте-Карло

900000

Кол-во сегментов отрезка для метода Симпсона

10





1.txt – Блокнот

---

Файл Правка Формат Вид Справка

---

$f(x)=x^2$   
Границы  
-1  
2  
Кол-во испытаний для метода Монте-Карло  
900000  
Кол-во сегментов отрезка для метода Симпсона  
10  
Симпсон: 3  
Монте-Карло: 2,99818288230642

С увеличением количества испытаний точность результата также увеличивается, но для точных результатов нужно очень много испытаний (даже 900000 испытаний дали результат с погрешностью примерно 0,002).

2)

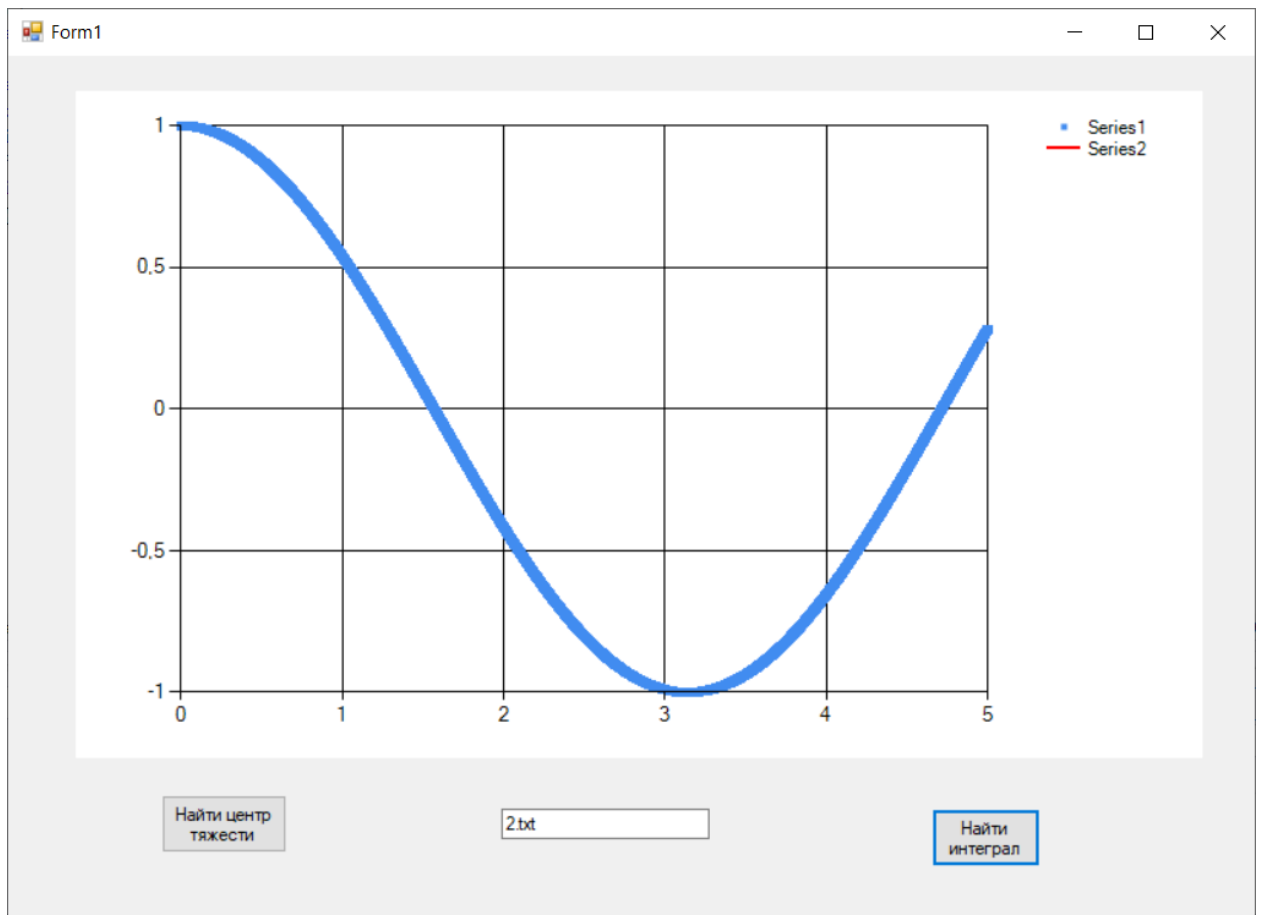
\*2.txt – Блокнот

---

Файл Правка Формат Вид Справка

---

$f(x)=\cos(x)$   
Границы  
0  
5  
Кол-во испытаний для метода Монте-Карло  
900000  
Кол-во сегментов отрезка для метода Симпсона  
10



2.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=\cos(x)$

Границы

0

5

Кол-во испытаний для метода Монте-Карло

900000

Кол-во сегментов отрезка для метода Симпсона

10

Симпсон: -0,958945240510608

Монте-Карло: -0,955395452611969

Увеличим кол-во испытаний

2.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x)=\cos(x)$

Границы

0

5

Кол-во испытаний для метода Монте-Карло

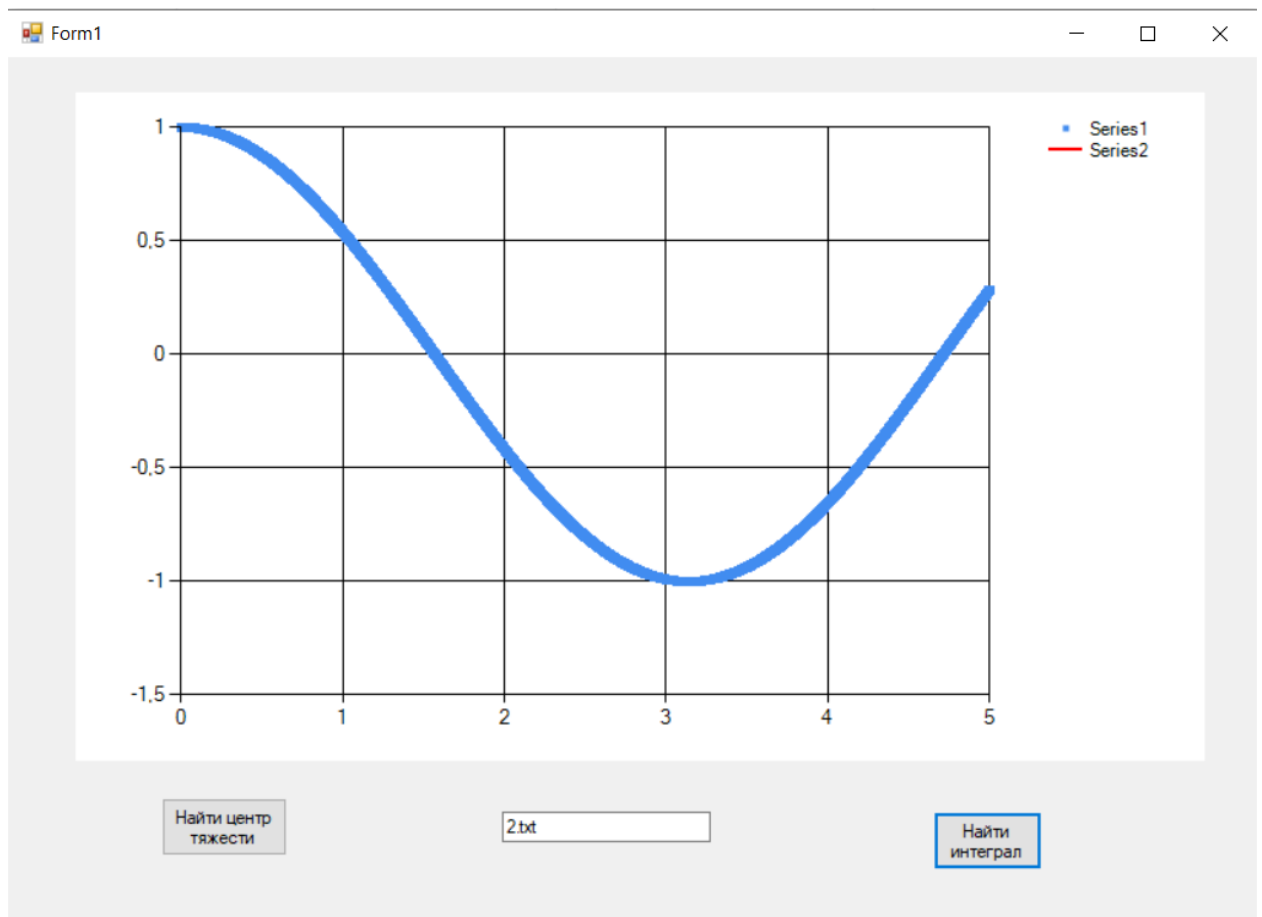
9000000

Кол-во сегментов отрезка для метода Симпсона

10

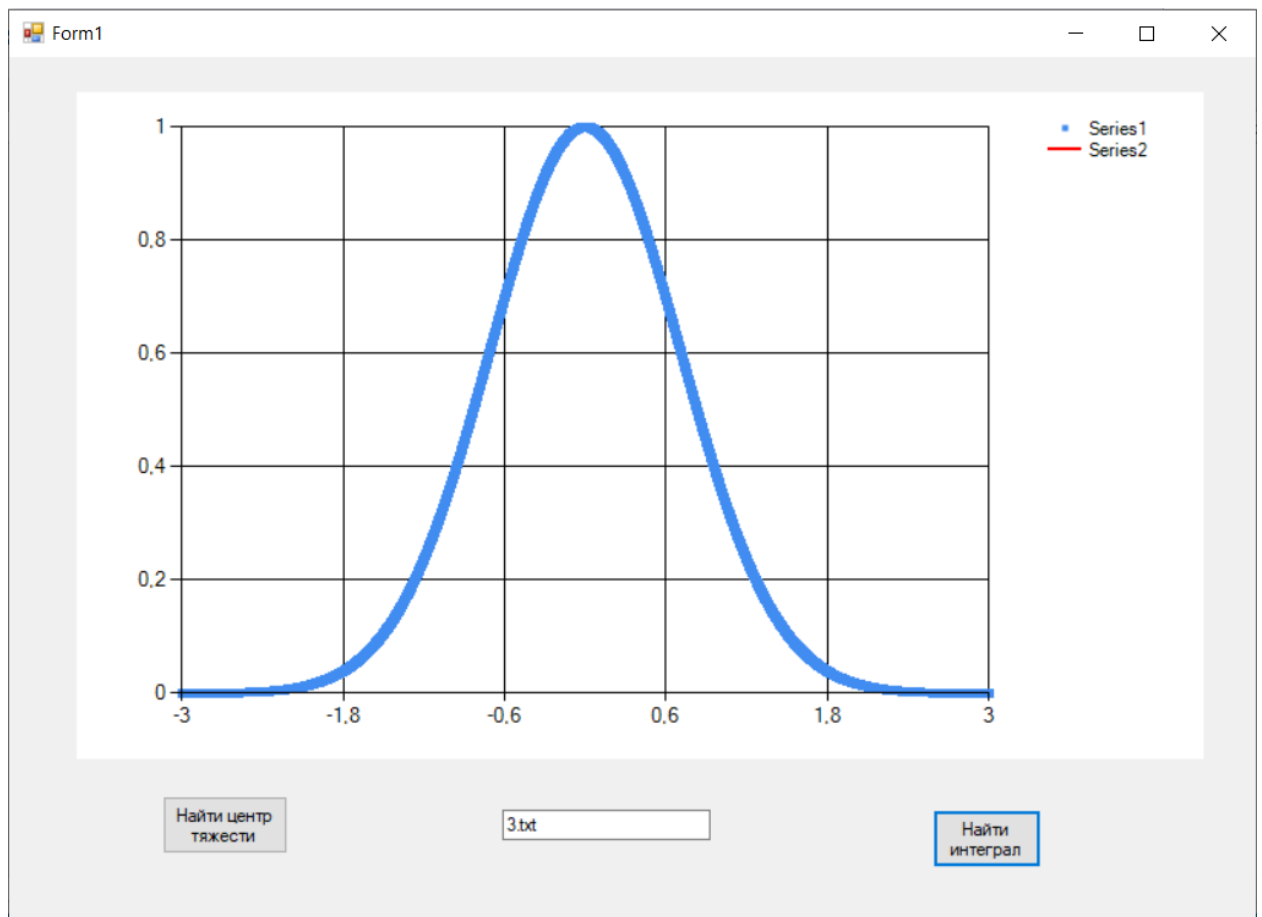
Симпсон: -0,958945240510608

Монте-Карло: -0,959223875264824



3)

```
*3.txt – Блокнот
Файл  Правка  Формат  Вид  Справка
f(x)=e^(-x*x)
Границы
-3
3
Кол-во испытаний для метода Монте-Карло
9000000
Кол-во сегментов отрезка для метода Симпсона
10
```



3.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x) = e^{(-x^2)}$

Границы

-3

3

Кол-во испытаний для метода Монте-Карло

9000000

Кол-во сегментов отрезка для метода Симпсона

10

Симпсон: 1,77241310449202

Монте-Карло: 1,77323488210687

Другой тест с таким же количеством испытаний

3.txt – Блокнот

Файл Правка Формат Вид Справка

$f(x) = e^{(-x \cdot x)}$

Границы

-3

3

Кол-во испытаний для метода Монте-Карло

9000000

Кол-во сегментов отрезка для метода Симпсона

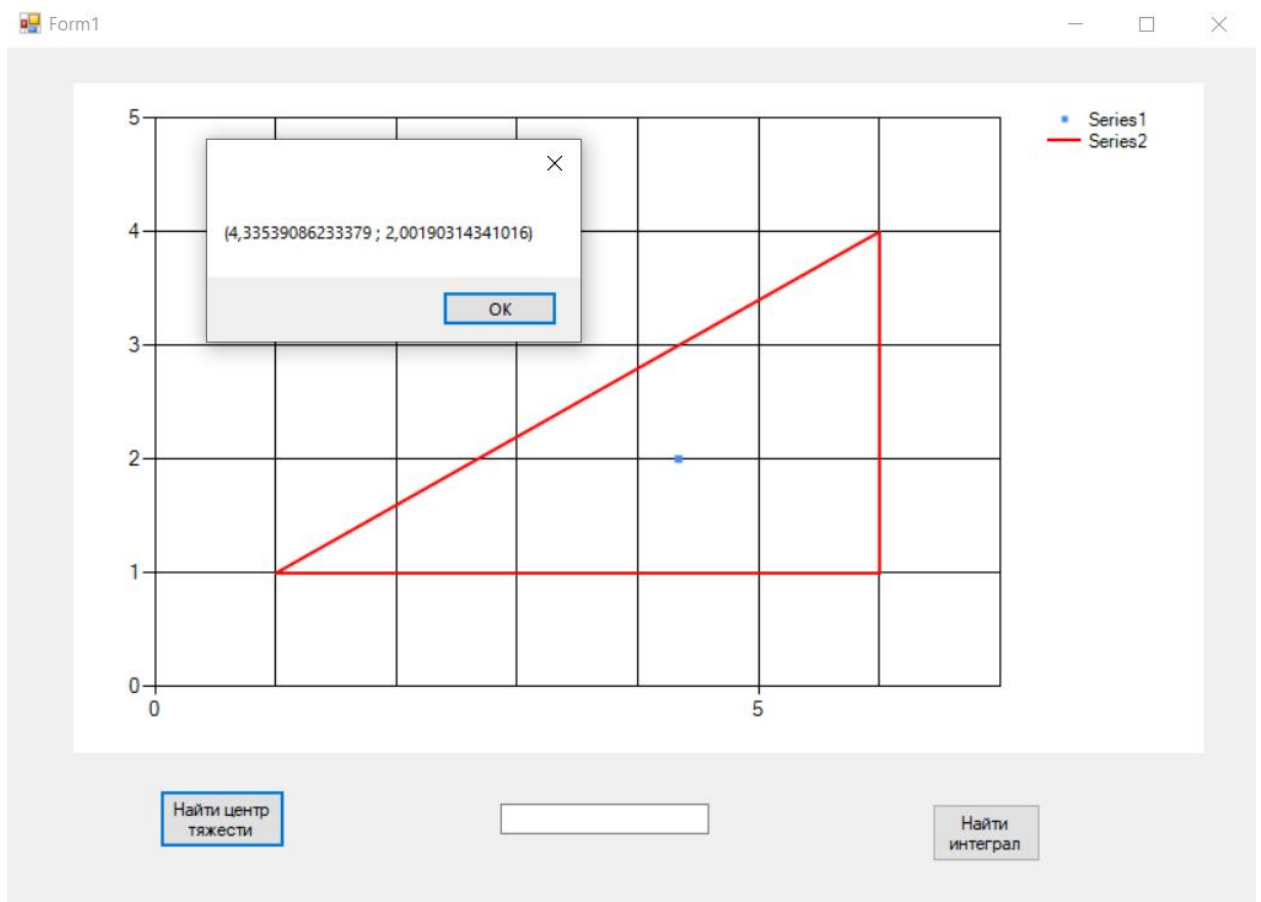
10

Симпсон: 1,77241310449202

Монте-Карло: 1,77192120117762

Результаты отличаются, потому что они зависят от выбранных случайно точек.

#### 4) Центр тяжести прямоугольного треугольника



Вывод: результат полученный методом Монте-Карло, помимо фактора случайности, зависит также от количества испытаний. Для более-менее высокой точности требуется очень много испытаний. Для того чтобы найти определенный интеграл с точностью  $\epsilon$  и доверительной вероятностью  $\gamma$  нужно с помощью функции Лапласа найти  $x$  удовлетворяющий  $\Phi(x) = \gamma$  и проделать такое кол-во испытаний, чтобы выполнялось

условие  $\chi^*(D/n)^{1/2} < \epsilon$  ( $D$  – дисперсия случайной величины определяемой в испытаниях,  
 $n$  – кол-во испытаний).