



Министерство науки и высшего образования Российской
Федерации
Федеральное государственное бюджетное образовательное
учреждение высшего образования
«Московский государственный технический университет имени
Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы управления»

КАФЕДРА «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ №1

По курсу: "Моделирование"

Студент _____ Сукочева Алис _____

Группа _____ ИУ7-73Б _____

Тема _____ Изучение функций распределения и плотности распределения _____

Студент: _____ Сукочева А. _____

подпись, дата _____ Фамилия, И.О.

Преподаватель: _____ Рудаков И.В. _____

подпись, дата _____ Фамилия, И. О.

1 Задание

Разработать программу для построения графиков распределений:

- а) равномерное распределение;
- б) нормальное распределение.

Также нужно реализовать графический интерфейс, который даст возможность выбора закона распределения и его параметров

2 Результат

На рисунках 2.1 и 2.2 приведены примеры работы программы.

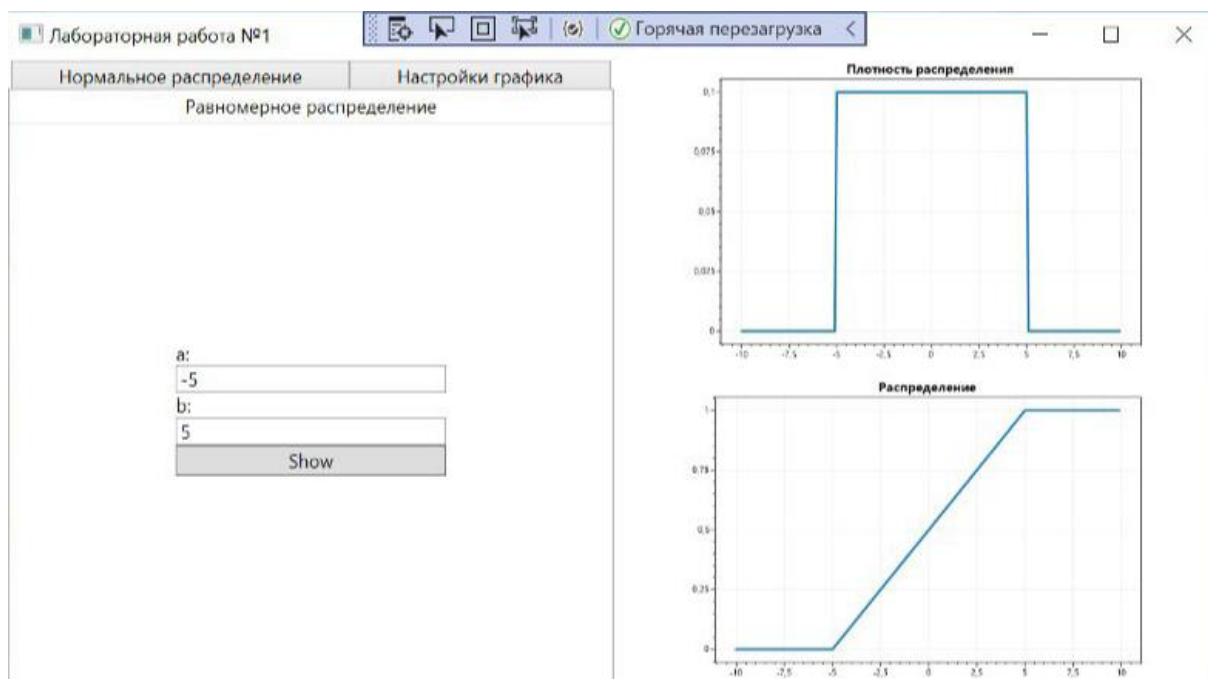


Рисунок 2.1 — Графики распределения и плотности распределения для равномерного распределения

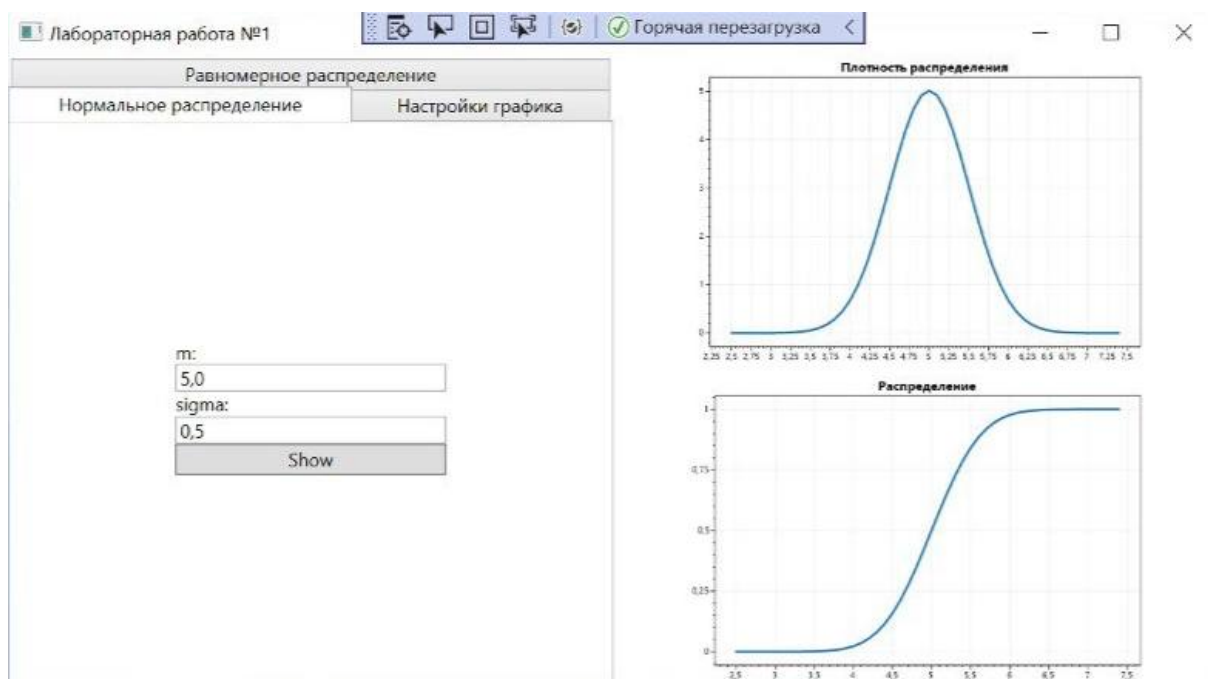


Рисунок 2.2 — Графики распределения и плотности распределения для нормального распределения

3 Текст программы

В листинге 3.1 представлен класс `Distribution`, отвечающий за вычисление значений функций, указанных в задании.

Листинг 3.1 — Class `Distribution`

```
1 public static class Distribution
2 {
3     public static double UniformDistribution(double a, double b, double x)
4     {
5         return a <= x && x <= b ? 1 / (b - a) : 0;
6     }
7
8     public static double UniformDistributionFunc(double a, double b,
9         double x)
10    {
11        if (x < a)
12        {
13            return 0;
14        }
15        if (x > b)
16        {
17            return 1;
18        }
19        return (x - a) / (b - a);
20    }
21
22    public static double NormalDistribution(double m, double sig, double x)
23    {
24        return 1 / sig * Math.Sqrt(2 * Math.PI) * Math.Pow(Math.E, -
25            Math.Pow(x - m, 2) / (2 * sig * sig));
26    }
27
28    public static double NormalDistributionFunc(double m, double sig,
29        double x)
30    {
31        return (1 + SpecialFunctions.Erf((x - m) / sig / Math.Sqrt(2))) /
32            2;
33    }
34
35    public static double[] Apply(double[] xs, double a, double b,
36        Func<double, double, double, double> f)
37    {
38        double[] ys = new double[xs.Length];
39
40        for (int i = 0; i < xs.Length; i++)
41        {
42            ys[i] = f(xs[i], a, b);
43        }
44    }
45 }
```

```
37         ys[i] = f(a, b, xs[i]);
38     }
39
40     return ys;
41 }
42 }
```