



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» (ИУ7)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.04 Программная инженерия

О Т Ч Е Т

по лабораторной работе № 1

Название: Изучение функций распределения и функций
плотности распределения случайных чисел.

Дисциплина: Моделирование

Вариант 3

Студент

ИУ7-72Б
(Группа)

(Подпись, дата)

Е.В. Брянская
(И.О. Фамилия)

Преподаватель

(Подпись, дата)

И.В. Рудаков
(И.О. Фамилия)

Москва, 2021

1 Задание

Построить график функций распределения и плотности для:

1. равномерного распределения;
2. по варианту:
 - 2.1. Пуассона;
 - 2.2. Эрланга;
 - 2.3. **нормального распределения**;
 - 2.4. распределение Гаусса.

И разработать соответствующий интерфейс.

2 Результаты выполнения работы

Равномерное распределение

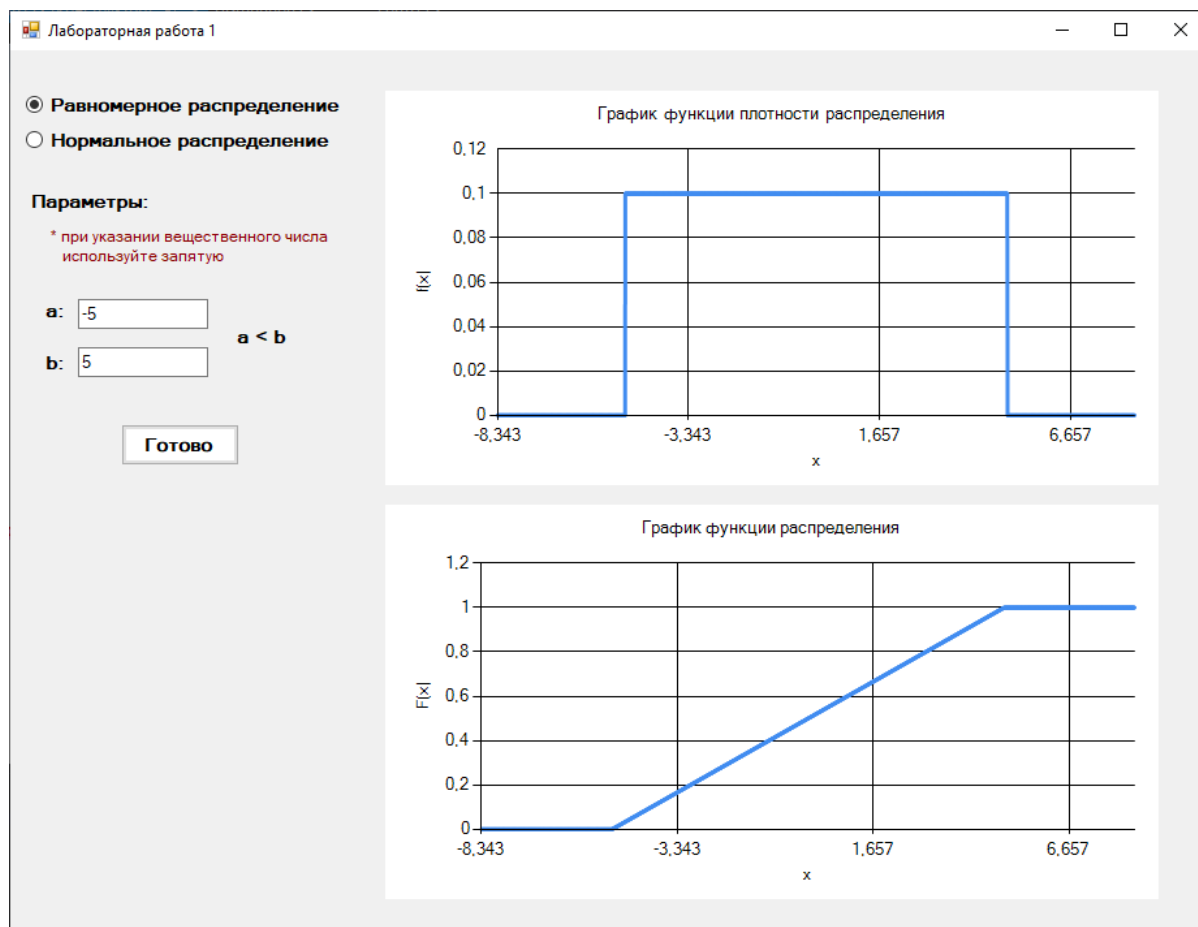


Рисунок 2.1 — Графики функции плотности распределения и функции распределения равномерной случайной величины при $a = -5$, $b = 5$.

Нормальное распределение

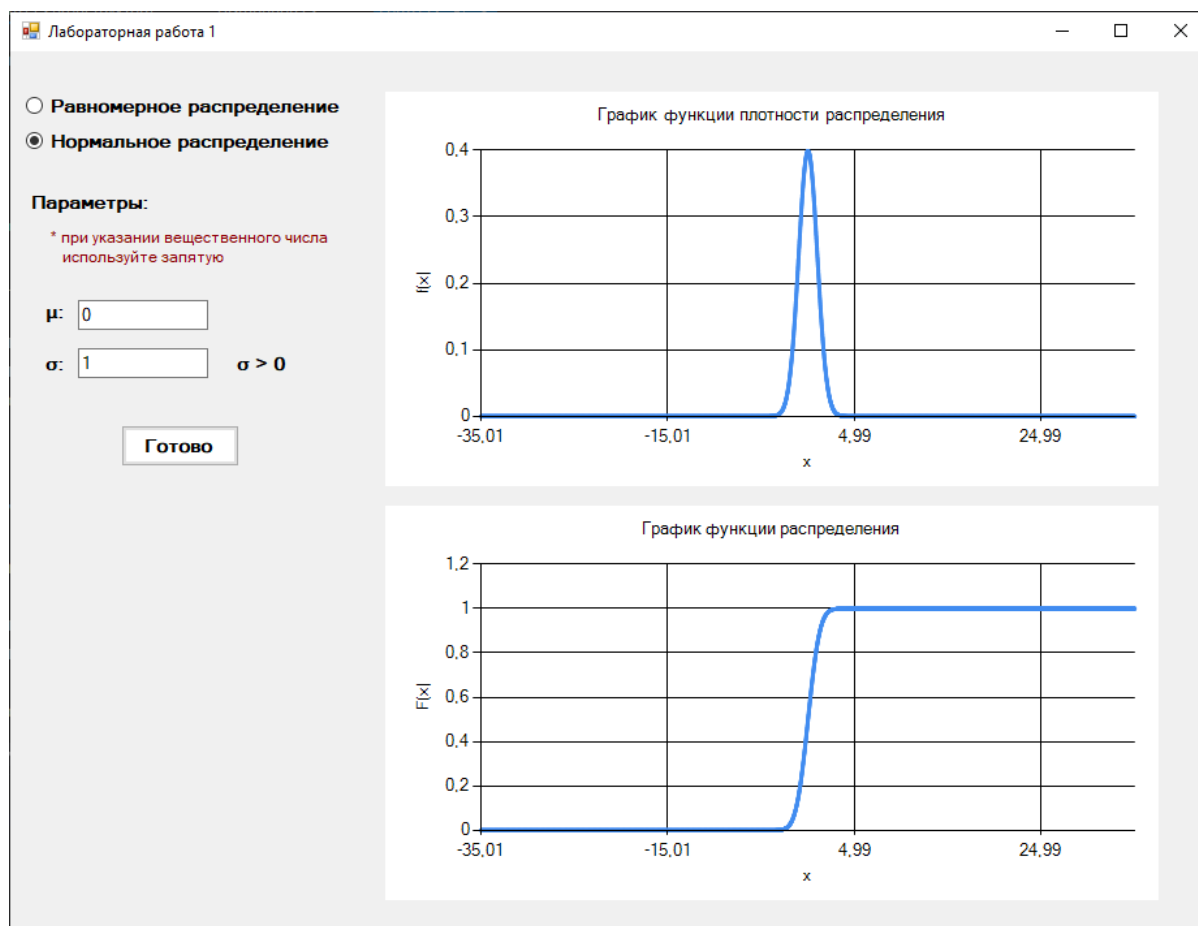


Рисунок 2.2 — Графики функции плотности распределения и функции распределения равномерной случайной величины при $\mu = 0$, $\sigma = 1$.

3 Код программы

Ниже представлены основные методы для расчётов.

```
1 class Distribution
2 {
3     public static void Equal(double a, double b, out double[] arrX, out double[] arrf, out
4         double[] arrF)
5     {
6         double d = b - a;
7         double step = d / 500;
8         double x = a - d / 3;
9
10        arrX = new double[(int)(5 * d / (3 * step)) + 1];
11        arrf = new double[(int)(5 * d / (3 * step)) + 1];
12        arrF = new double[(int)(5 * d / (3 * step)) + 1];
13
14        if (a >= b)
15            throw new Exception();
16
17        for (int i = 0; i < arrX.Length; i++)
18        {
19            arrX[i] = Math.Round(x, 3);
20            arrf[i] = _Equalf(a, b, x);
21            arrF[i] = _EqualF(a, b, x);
22            x += step;
23        }
24
25        private static double _Equalf(double a, double b, double x)
26        {
27            return (a <= x && x <= b) ? 1 / (b - a) : 0;
28        }
29
30        private static double _EqualF(double a, double b, double x)
31        {
32            if (x < a) return 0;
33            if (x > b) return 1;
34            return (x - a) / (b - a);
35        }
36
37        public static void Normal(double mu, double sigma, out double[] arrX, out double[] arrf,
38            out double[] arrF)
39        {
40            double a = -35, b = 35;
41            double step = 1e-2;
42            double x = a;
43
44            arrX = new double[(int)((b - a) / step) + 1];
45            arrf = new double[(int)((b - a) / step) + 1];
46            arrF = new double[(int)((b - a) / step) + 1];
```

```

46
47     for (int i = 0; i < arrX.Length; i++)
48     {
49         arrX[i] = Math.Round(x, 3);
50         arrf[i] = _Normalf(mu, sigma, x);
51         arrF[i] = _NormalF(mu, sigma, x);
52         x += step;
53     }
54 }
55
56 private static double _Normalf(double mu, double sigma, double x)
57 {
58     double pi = 3.14;
59
60     return 1 / (sigma * Math.Sqrt(2 * pi)) * Math.Exp(-Math.Pow(x - mu, 2) / (2 * sigma *
        sigma));
61 }
62
63 private static double _NormalF(double mu, double sigma, double x)
64 {
65     return 0.5 * (1 + SpecialFunctions.Erf((x - mu) / (sigma * Math.Sqrt(2))));
66 }
67 }

```