

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 09.03.01 Информатика и вычислительная техника

ОТЧЕТ

по лабораторной работе № __3__

Гема: Генераторы псевдослучайных последовательностей			
Дисциплина:	<u>Моделирование</u>		
Студент	_ИУ7-72Б		В.А. Иванов
	(Группа)	(Подпись, дата)	(И.О. Фамилия)
Преподаватель			И.В. Рудаков
		(Подпись, дата)	(И.О. Фамилия)

Москва, 2021

1. Задание

Используя алгоритмический и табличный метод сгенерировать последовательность из 1000 одно-, дву- и трёхразрядных чисел. Отобразить первые 10 в графическом интерфейсе. Составить статистический критерий оценки случайности последовательностей и вывести его значение для каждой последовательности. Также предусмотреть возможность получения оценки для 10 чисел введённых пользователем.

2. Результаты

2.1. Теория

Генераторы случайных чисел разделяются на:

- аппаратные
- табличные
- алгоритмические

В данной лабораторной работе реализованы два последних из них

2.1.1. Алгоритмический генератор

Алгоритмический генератор использует одно или несколько значений для вычисления нового числа, после чего также используется для генерации последующих чисел.

В данной работе был использован линейный конгруэнтный метод. Последовательность в нём вычисляется по формуле

$$X_{n+1} = (aX_n + c) \bmod m, \ n \ge 1$$
 (2.1)

В данной лабораторной использованы значения из минимального стандартного генератора случайных чисел: a=16807, c=2147483647, $m=MAX_INT$. В качестве начального значения X_1 используется системное время.

2.1.2. Табличный генератор

Табличный генератор использует заранее подготовленные последовательности хранящиеся в памяти компьютера.

В лабораторной работе файлы были сгенерированы при помощи стандартной библиотеки языка C#. Начальное значение генерируется так же и обозначает смещение относительно начала файла.

2.1.3. Критерий оценки

В качестве статистического критерия оценки случайности был выбран критерий "хи-квадрат". В первую очередь вычисляется статистика V

$$V = \frac{1}{n} \sum_{s=min}^{max} (\frac{Y_i^2}{p}) - n \tag{2.2}$$

где n - длина последовательности, [min, max) - диапазон чисел, Y_i - количество повторений числа i, p = 1/(max - min).

После этого вычисляется критерий - квантиль хи-квадрат (степень свободы max - min - 1) от значения V. Полученное значение предлагается интерпретировать следующим образом:

- < 0.1 или > 0.9 последовательность считается недостаточно случайной.
- > 0.1 и < 0.9 последовательность считается случайной.

2.2. Работа программы

Для демонстрации работы критерия предоставленны 10 полей ввода. Оценка производятся для целых чисел от 0 до 9. Примеры работы программы приведены на рисунках 2.1, 2.2 и 2.3.



Рис. 2.1 — Введена последовательность из одинаковых чисел

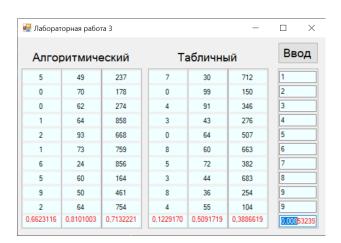


Рис. 2.2 — Введена последовательность из разных чисел

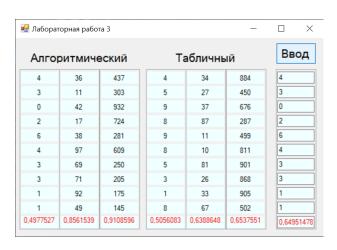


Рис. 2.3 — Введена случайная последовательность

3. Текст программы

В листинге 3.1 представлен фрагмент кода программы, отвечающий за генерирование последовательностей и вычисление критерия.

Листинг 3.1 — Вычисление функций распредлений

```
class RndAlgGen: IRndGen
2
  {
3
    private int seed;
     public RndAlgGen(int minN, int maxN, int s=0)
4
5
6
       MinN = minN;
7
       MaxN = maxN;
       SetSid(s);
8
9
    }
10
     public override void SetSid(int s)
11
12
     {
13
       seed = s;
14
15
     public override int Rand()
16
17
18
       seed = seed * 16807 + 2147483647;
       var res = ((uint)seed >> 16) \% (MaxN - MinN) + MinN;
19
       return (int) res;
20
21
    }
22
     public override int[] RandArr(int size)
23
24
    {
25
       int[] res = new int[size];
       for (int i = 0; i < size; i++)
26
27
       res[i] = Rand();
       return res;
28
29
     }
30 }
```

```
31
32 class RndTabGen : IRndGen
33 {
     private int seed;
34
     private StreamReader f;
35
     private string fpath;
36
37
     public static void FillFile(string fpath_, int min, int max
38
     )
    {
39
       var rand = new Random();
40
       using (StreamWriter sw = File .CreateText(fpath_))
41
42
         for (int i = 0; i < (max - min) * 1000; i++)
43
         sw.WriteLine(rand.Next(min, max));
44
       }
45
46
     public RndTabGen(int minN, int maxN, string fp, int s = 0)
47
48
    {
       MinN = minN;
49
       MaxN = maxN;
50
       fpath = fp;
51
       SetSid(s);
52
       var chi = new ChiSquared(1.0);
53
54
    }
55
     private void OpenFile(string fp)
56
57
58
       fpath = fp;
       if (f != null)
59
       f. Close();
60
       f = File.OpenText(fpath);
61
62
       for (int i = 0; i < seed; i++)
       f ReadLine();
63
```

```
}
64
65
     public override void SetSid(int s)
66
67
       seed = s;
68
       OpenFile (fpath);
69
     }
70
71
     public override int Rand()
72
73
     {
       string fileStr = f.ReadLine();
74
       if (fileStr == null)
75
76
         SetSid(0);
77
         fileStr = f.ReadLine();
78
79
       }
80
       seed++;
81
       var res = (UInt32.Parse(fileStr)) % (MaxN - MinN) + MinN;
82
       return (int)res;
83
84
     }
85
     public override int[] RandArr(int size)
86
87
     {
       int[] res = new int[size];
88
       for (int i = 0; i < size; i++)
89
       res[i] = Rand();
90
91
       return res;
    }
92
93 | }
94
95
96 class Crit
97 {
```

```
public int[] arr;
98
     public Crit(int[] arr )
99
100
101
        arr = arr;
     }
102
103
104
      public double Value(int min, int max)
105
106
        int n = arr Length;
        double p = 1.0 / (max - min);
107
        double acc = 0;
108
        for (int i = min; i < max; i++)
109
        acc += Math.Pow(arr.Count(x \Rightarrow x == i), 2) / p;
110
        acc = acc / (double)n - n;
111
112
        double res = ChiSquared.CDF(max - min - 1, acc);
113
114
        return res;
     }
115
116 }
```