



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 работе №3

Название Исследование псевдослучайных чисел

Дисциплина Моделирование

Студент Прохорова Л. А.

Группа ИУ7-73Б

Оценка (баллы) _____

Преподаватель Рудаков И. В.

Москва — 2022 г.

1 Задание

Изучить методы генерирования псевдослучайных чисел, а также критерии оценки случайности последовательности. Реализовать критерий оценки случайной последовательности. Сравнить результаты работы данного критерия на одноразрядных, двухразрядных и трехразрядных последовательностях целых чисел. Последовательности получать алгоритмическим и табличным способами. Предусмотреть возможность ввода десяти чисел для оценки их по критерию.

2 Теоретическая часть

Случайные числа — искусственно полученная последовательность реализаций случайной величины с заданным законом распределения.

На практике используется три основных способа получения случайных чисел:

- Аппаратный
- Табличный(файловый)
- Алгоритмический(программный)

Аппаратный способ получения случайных чисел

Аппаратные генераторы случайных чисел — это устройства, использующие для создания случайных чисел замеры параметров некоторых физических процессов. Как правило, аппаратный генератор случайных чисел состоит из источника энтропии и устройства, преобразующего значения, полученные с источника энтропии, в нужный формат.

Табличный способ получения случайных чисел

В данном способе в качестве источника случайных чисел используют заранее подготовленные таблицы, содержащие проверенные некоррелированные числа. Недостатки такого способа: использование внешнего ресурса для хранения чисел, ограниченность последовательности, предопределенность значений.

Алгоритмический способ получения случайных чисел

Алгоритмический генератор является комбинацией физического генератора и детерминированного алгоритма. Такой генератор использует ограниченный набор данных, полученный с выхода физического генератора для создания длинной последовательности чисел преобразованиями исходных чисел. Из-за дороговизны аппаратных генераторов случайных чисел в большинстве случаев, в качестве источника энтропии используются ресурсы вычислительной машины, на которой выполняется программа генерации ПСЧ. При отсутствии аппаратного генератора случайных чисел в качестве источника энтропии могут использоваться:

- состояние системных часов;
- время задержек между нажатиями клавиш клавиатуры или движениями мышки;
- содержимое буферов ввода/вывода;
- значения, получаемые при работе системы.

2.1 Выбранные методы

Для получения случайных чисел алгоритмическим способом выбран линейный конгруэнтный метод.

Линейный конгруэнтный метод

Для осуществления генерации чисел данным методом, необходимо задать 4 числа:

$m > 0$, модуль

Последовательность случайных чисел генерируется при помощи формулы:

(1)

При некоторых наборах чисел m , a , c , и X_0 последовательность не может быть "случайной". Поэтому важно правильно их подобрать. В конгруэнтной последовательности всегда существуют циклы - периоды, необходимо чтобы последовательность, которую мы используем, имела относительно длинный период.

Выбранный критерий оценки случайной последовательности - критерий "хи-квадрат". Это один из самых известных статистических критериев, также это основной метод, используемый в сочетании с другими критериями.

С помощью этого критерия можно узнать, удовлетворяет ли генератор случайных чисел требованию равномерного распределения или нет. Для оценки по этому критерию необходимо вычислить статистику V по формуле:

$$V = \frac{1}{n} \sum_{s=1}^k \left(\frac{Y_s^2}{p_s} \right) - n \quad (2)$$

где n – количество независимых испытаний, k – количество категорий, Y_s – число наблюдений, которые действительно относятся к категории S , p_s – вероятность того, что каждое наблюдение относится к категории s .

Значение V является значением критерия «хи-квадрат» для экспериментальных данных. Приемлемое значение этого критерия можно определить по таблице 1. Для этого используем строку с $v = k-1$, где $k = 10, 90, 900$ для задания лабораторной. P в этой таблице – это вероятность того, что экспериментальное значение $V_{\text{эксп.}}$ будет меньше табулированного

(теоретического) $V_{\text{теор.}}$ или равно ему. Ее также можно рассматривать как доверительную вероятность.

Если вычисленное V окажется меньше 1% точки или больше 99% точки, можно сделать вывод, что эти числа недостаточно случайные. Если V лежит между 1% и 5% точками или между 95% и 99% точками, то эти числа «подозрительны». Если V лежит между 5% и 10% точками или 90%-95% точками, то числа можно считать «почти подозрительными». Проверка по "хи-квадрат" критерию часто производится три раза и более с разными данными. Если по крайней мере два из трех результатов оказываются подозрительными, то числа рассматриваются как недостаточно случайные.

Таблица 1
НЕКОТОРЫЕ ПРОЦЕНТНЫЕ ТОЧКИ χ^2 -РАСПРЕДЕЛЕНИЯ

	$p = 1\%$	$p = 5\%$	$p = 25\%$	$p = 50\%$	$p = 75\%$	$p = 95\%$	$p = 99\%$
$\nu = 1$	0.00016	0.00393	0.1015	0.4549	1.323	3.841	6.635
$\nu = 2$	0.02010	0.1026	0.5754	1.386	2.773	5.991	9.210
$\nu = 3$	0.1148	0.3518	1.213	2.366	4.108	7.815	11.34
$\nu = 4$	0.2971	0.7107	1.923	3.357	5.385	9.488	13.28
$\nu = 5$	0.5543	1.1455	2.675	4.351	6.626	11.07	15.09
$\nu = 6$	0.8721	1.635	3.455	5.348	7.841	12.59	16.81
$\nu = 7$	1.239	2.167	4.255	6.346	9.037	14.07	18.48
$\nu = 8$	1.646	2.733	5.071	7.344	10.22	15.51	20.09
$\nu = 9$	2.088	3.325	5.899	8.343	11.39	16.92	21.67
$\nu = 10$	2.558	3.940	6.737	9.342	12.55	18.31	23.21
$\nu = 11$	3.053	4.575	7.584	10.34	13.70	19.68	24.72
$\nu = 12$	3.571	5.226	8.438	11.34	14.85	21.03	26.22
$\nu = 15$	5.229	7.261	11.04	14.34	18.25	25.00	30.58
$\nu = 20$	8.260	10.85	15.45	19.34	23.83	31.41	37.57
$\nu = 30$	14.95	18.49	24.48	29.34	34.80	43.77	50.89
$\nu = 50$	29.71	34.76	42.94	49.33	56.33	67.50	76.15
$\nu > 30$	$\nu + \sqrt{2\nu}x_p + \frac{2}{3}x_p^2 - \frac{2}{3} + O(1/\sqrt{\nu})$						
$x_p =$	-2.33	-1.64	-0.674	0.00	0.674	1.64	2.33

Рисунок 1 – Некоторые процентные точки "хи-квадрат"распределения (Источник: Кнут Д. Э. «Искусство программирования»).

k - 1	p = 1%	p = 5%	p = 25%	p = 50%	p = 75%	p = 95%	p = 99%
9	2.088	3.325	5.899	8.343	11.39	16.92	21.67
89	60.93	68.25	79.68	88.33	97.60	112.02	122.94
899	803.31	830.41	870.05	898.33	927.23	969.86	1000.57

Таблица 1 – Таблица значений $V_{\text{теор}}$ для количества степеней свободы по заданию

3 Результаты работы программы

Программа, реализованная в лабораторной работе, выводит на экран таблицу из 7 столбцов и 13 строк. 10 строк представлены для того, чтобы можно было пронаблюдать, какие числа возвращает генератор случайных чисел. Для каждого из реализованных методов в таблице есть по три столбца для чисел с разным количеством разрядов. В строке "коэффициент" выводится значение V , подсчитанное для каждого столбца ($N = 10000$). В последней строке выводится заключение, сделанное в результате анализа вычисленного значения по таблице 1.

Так же есть возможность ввести собственную последовательность и проанализировать ее.

На рисунках 2, 3, 4 приведен пример трех запусков программ.

Табличный метод				Алгоритмический метод			
№	1 разряд	2 разряда	3 разряда	1 разряд	2 разряда	3 разряда	
0	1	58	298	3	51	977	
1	2	51	588	6	92	500	
2	1	63	888	7	31	223	
3	3	77	252	8	84	122	
4	4	99	285	5	19	673	
5	1	78	532	4	20	180	
6	1	68	722	7	19	695	
7	4	72	105	6	74	986	
8	4	21	121	5	99	125	
9	1	42	724	8	34	212	
Коэффициент	10.26599999999622	111.87800000000061	801.2600000000002	7.727999999999156	74.36599999999999	899.1800000000003	
Критерий	Числа случайные	Числа случайные	Числа не случайные	Числа случайные	Числа случайные	Числа случайные	

Введите 1 если хотите проанализировать свою последовательность: 1
 Выберите размерность вводимой последовательности
 Одноразрядные - введите 1, двухразрядные - введите 2, трехразрядные - введите 3: 1
 Введите последовательность чисел (через пробел)
 1 2 3 4 5 6 7 8 9
 Коэффициент: 1.0
 Числа не случайные

Рисунок 2 – Результат работы первого запуска программы

Табличный метод				Алгоритмический метод			
№	1 разряд	2 разряда	3 разряда	1 разряд	2 разряда	3 разряда	
0	9	44	900	3	51	977	
1	3	13	570	6	92	500	
2	0	38	896	7	31	223	
3	1	19	494	8	84	122	
4	7	48	931	5	19	673	
5	8	84	613	4	20	180	
6	7	29	308	7	19	695	
7	8	23	829	6	74	986	
8	2	67	742	5	99	125	
9	8	39	709	8	34	212	
Коэффициент	13.572000000000116	78.79399999999987	832.2199999999993	7.727999999999156	74.36599999999999	899.1800000000003	
Критерий	Числа случайные	Числа случайные	Числа случайные	Числа случайные	Числа случайные	Числа случайные	

Введите 1 если хотите проанализировать свою последовательность: 1
 Выберите размерность вводимой последовательности
 Одноразрядные - введите 1, двухразрядные - введите 2, трехразрядные - введите 3: 2
 Введите последовательность чисел (через пробел)
 10 40 30 80 70 11 13 03 04
 Коэффициент: 81.0
 Числа случайные

Рисунок 3 – Результат работы второго запуска программы

Табличный метод				Алгоритмический метод			
№	1 разряд	2 разряда	3 разряда	1 разряд	2 разряда	3 разряда	
0	3	93	344	3	51	977	
1	8	67	437	6	92	500	
2	7	74	414	7	31	223	
3	8	90	998	8	84	122	
4	7	72	214	5	19	673	
5	0	83	462	4	20	180	
6	1	63	787	7	19	695	
7	0	72	948	6	74	986	
8	5	70	191	5	99	125	
9	6	52	500	8	34	212	
Кoeffициент	5.373999999999796	62.486000000000786	831.8600000000006	7.727999999999156	74.36599999999999	899.1800000000003	
Критерий	Числа случайные	Числа подозрительные	Числа случайные	Числа случайные	Числа случайные	Числа случайные	

Введите 1 если хотите проанализировать свою последовательность: 1
 Выберите размерность вводимой последовательности
 Одноразрядные - введите 1, двухразрядные - введите 2, трехразрядные - введите 3: 3
 Введите последовательность чисел (через пробел)
 124 521 489 377 189 112 379 937 589 444 387 111
 Коэффициент: 888.0
 Числа случайные

Рисунок 4 – Результат работы третьего запуска программы

Проанализируем три выведенных результата для каждого столбца для того чтобы сделать вывод об эффективности генератора случайных чисел.

Табличный способ получения случайных чисел

- Одноразрядные числа - во всех трех экспериментах числа в последовательности случайные, соответственно одноразрядные числа, генерируемые с помощью табличного метода являются случайными при оценке критерием "хи-квадрат".
- Двухразрядные числа - в двух экспериментах числа в последовательности случайные, в одном подозрительные. Так как количество экспериментов с ответом что числа подозрительные не 2 и не 3, то можно считать, что двухразрядные числа, генерируемые с помощью табличного метода являются случайными при оценке критерием "хи-квадрат".
- Трехразрядные числа - в двух экспериментах числа в последовательности случайные, в одном не являются случайными. Так как количество экспериментов с ответом что числа не случайные только 1 из 3, то можно считать это некоторой случайностью при генерации, которая не является закономерной. (После данного эксперимента было

проведено еще 20 экспериментов которые не отражены в отчете и каждый раз получался ответ - числа случайные). Можно считать что трехразрядные числа, генерируемые с помощью табличного метода являются случайными при оценке критерием "хи-квадрат".

Алгоритмический способ получения случайных чисел

Во всех трех экспериментах и для одnorазрядных, и для двухразрядных, и для трехразрядных чисел было установлено, что они случайные.

Помимо таблиц на рисунках 2, 3, 4 представлен пример обработки последовательности, введенной пользователем.

Так как иногда числа, сгенерированные с помощью табличного способа получения случайных чисел оказывались не случайными или подозрительными, то можно считать что алгоритмический способ получения случайных чисел эффективнее.

4 Код программы

Программа разработана в интегрированной среде разработки для языка программирования Python - PyCharm. В листинге 1 приведена реализация лабораторной работы.

```
0 from prettytable import PrettyTable
1 from itertools import islice
2
3 COUNT = 10000
4 m = 2. ** 31
5 a = 1664525
6 c = 1013904223
7
8 theor_koef_one_digit = {'1': 2.088, '5': 3.325, '25': 5.899, '50': 8.343,
9                          '75': 11.39, '95': 16.92, '99': 21.67}
10 theor_koef_two_digits = {'1': 60.93, '5': 68.25, '25': 79.68, '50': 88.33,
11                           '75': 97.60, '95': 112.02, '99': 122.94}
```

```

10 theor_koef_three_digits = {'1': 803.31, '5': 830.41, '25': 870.05, '50':
    898.33, '75': 927.23, '95': 969.86, '99': 1000.57}
11 theor_koefs = {'one_digit': theor_koef_one_digit, 'two_digits':
    theor_koef_two_digits, 'three_digits': theor_koef_three_digits}
12
13 class RandomGenerator:
14     def __init__(self):
15         self.current = 10
16
17     def get_random_number(self, low, high):
18         self.current = (a * self.current + c) % m
19         result = int(low + self.current % (high - low))
20         return result
21
22 def table_rand():
23     numbers = set()
24     with open('digits.txt') as file:
25         line_num = 0
26         lines = islice(file, line_num, None)
27         for l in lines:
28             numbers.update(set(l.split(" ")[1:-1]))
29             line_num += 1
30             if len(numbers) >= 3 * COUNT + 1:
31                 break
32         numbers.remove("")
33         numbers = list(numbers)[:3 * COUNT]
34     one_digit = [int(i) % 10 for i in numbers[:COUNT]]
35     two_digits = [int(i) % 90 + 10 for i in numbers[COUNT:COUNT * 2]]
36     three_digits = [int(i) % 900 + 100 for i in numbers[COUNT * 2:3 *
COUNT]]
37     return one_digit, two_digits, three_digits
38
39
40 def alg_rand():
41     random = RandomGenerator()
42     one_digit = [random.get_random_number(0, 10) for i in range(COUNT)]
43     two_digits = [random.get_random_number(10, 100) for i in range(COUNT)]
44     three_digits = [random.get_random_number(100, 1000) for i in range(
COUNT)]
45     return one_digit, two_digits, three_digits

```

```

46
47
48 def calc_hi(arr, start, end):
49     n = len(arr)
50     tab = [0 for i in range(start + end)]
51     for i in range(n):
52         tab[arr[i]] += 1
53     s = 0
54     for i in tab:
55         s += i * i
56     return s * (end - start) / n - n
57
58 def check_with_criterion(prac_koef, digit_str):
59     if prac_koef < theor_koefs[digit_str]['1'] or prac_koef > theor_koefs[
digit_str]['99']:
60         return "Числа не случайные"
61
62     if prac_koef >= theor_koefs[digit_str]['1'] and prac_koef <=
theor_koefs[digit_str]['5']:
63         return "Числа подозрительные"
64
65     if prac_koef <= theor_koefs[digit_str]['99'] and prac_koef >=
theor_koefs[digit_str]['95']:
66         return "Числа подозрительные"
67     return "Числа случайные"
68 def main():
69     numbers = [i for i in range(10)]
70     table_tbl = PrettyTable()
71     one_tbl, two_tbl, three_tbl = table_rand()
72
73     table_tbl.add_column("№", numbers)
74     table_tbl.add_column('1 разряд', one_tbl[:10])
75     table_tbl.add_column('2 разряда', two_tbl[:10])
76     table_tbl.add_column('3 разряда', three_tbl[:10])
77
78     one_alg, two_alg, three_alg = alg_rand()
79     table_tbl.add_column('1 разряд', one_alg[:10])
80     table_tbl.add_column('2 разряда', two_alg[:10])
81     table_tbl.add_column('3 разряда', three_alg[:10])
82

```



```

117         except:
118             print("Некорректное значение")
119             return
120         arr.append(d)
121     if digit_str == '1':
122         hi_koef = calc_hi(arr, 0, 10)
123     elif digit_str == '2':
124         hi_koef = calc_hi(arr, 10, 100)
125     else:
126         hi_koef = calc_hi(arr, 100, 1000)
127     print("Коэффициент: ", hi_koef)
128     s = ['one_digit', 'two_digits', 'three_digits']
129     print(check_with_criterion(hi_koef, s[int(digit_str) - 1]))
130 else:
131     print("Некорректный ввод")
132     return
133
134 if __name__ == '__main__':
135     main()

```