



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»

ЛАБОРАТОРНАЯ РАБОТА №1

Дисциплина	<u>Моделирование</u>
Тема	<u>Генераторы псевдослучайных чисел</u>
Студент	<u>Набиев Ф.М.</u>
Группа	<u>ИУ7-73Б</u>
Оценка (баллы)	<u> </u>
Преподаватель	<u>Рудаков И.В.</u>

Москва, 2020 г.

1 Условие

Изучить и реализовать генератор псевдослучайных чисел программным и табличным методом. Получить 1, 2 и 3-х разрядные числа. Сравнить методы по определённому критерию и сделать выводы.

2 Теоретическая часть

В настоящей лабораторной работе рассматриваются программный и табличный методы генерации случайных чисел.

Программный генератор формирует псевдослучайные числа. Каждое последующее число в такой последовательности зависит от предыдущего.

Табличный генератор использует таблицу проверенных некоррелированных цифр в качестве источника случайных чисел.

2.1 Критерий случайности

Для сравнения описанных методов генерации случайных чисел воспользуемся **критерием частотности**, который позволяет определить равномерность сгенерированных чисел.

Определяется количество чисел на интервале $(\mu - \sigma, \mu + \sigma)$, где μ — математическое ожидание, а σ — среднеквадратичное отклонение. Идеальным результатом будем считать отношение длины рассматриваемого интервала к длине всего промежутка, на котором генерируется последовательность. Под полученным результатом будем понимать отношение количества сгенерированных чисел на интервале к количеству всех сгенерированных чисел.

3 Практическая часть

Рассмотрим результаты выполнения реализованной программы.

3.1 Результаты работы

```
~/Documents/Repositories/bmstu/MathModeling/sem_02/lab_01
> ./run.py 10
```

Программный метод:

No	1 разр.	2 разр.	3 разр.
1	9.0000	59.0000	418.0000
2	7.0000	89.0000	738.0000
3	4.0000	82.0000	138.0000
4	8.0000	77.0000	192.0000
5	7.0000	54.0000	700.0000
6	9.0000	35.0000	655.0000
7	2.0000	82.0000	310.0000
8	1.0000	20.0000	348.0000
9	3.0000	83.0000	626.0000
10	10.0000	30.0000	843.0000
Ожидаемый	0.6132	0.5341	0.5192
Полученный	0.7000	0.6000	0.6000

Табличный метод:

No	1 разр.	2 разр.	3 разр.
1	3.0000	62.0000	226.0000
2	2.0000	16.0000	707.0000
3	0.0000	22.0000	133.0000
4	3.0000	88.0000	368.0000
5	6.0000	72.0000	547.0000
6	4.0000	53.0000	931.0000
7	3.0000	66.0000	306.0000
8	0.0000	96.0000	277.0000
9	7.0000	56.0000	502.0000
10	0.0000	55.0000	225.0000
Ожидаемый	0.4630	0.5312	0.5261
Полученный	0.5000	0.6000	0.7000

Рис. 3.1 – Пример работы для 10 чисел

```
~/Documents/Repositories/bmstu/MathModeling/sem_02/lab_01
> ./run.py 100
```

Программный метод:

No	1 разр.	2 разр.	3 разр.
1	8.0000	84.0000	248.0000
2	3.0000	87.0000	967.0000
3	4.0000	46.0000	321.0000
4	2.0000	24.0000	642.0000
5	4.0000	47.0000	234.0000
...			
96	3.0000	49.0000	799.0000
97	7.0000	78.0000	317.0000
98	2.0000	10.0000	409.0000
99	8.0000	21.0000	220.0000
100	1.0000	72.0000	192.0000

Ожидаемый	0.5473	0.5869	0.6004
Полученный	0.4900	0.5400	0.5400

Табличный метод:

No	1 разр.	2 разр.	3 разр.
1	3.0000	45.0000	575.0000
2	7.0000	86.0000	682.0000
3	8.0000	71.0000	300.0000
4	0.0000	95.0000	449.0000
5	4.0000	91.0000	323.0000
...			
96	6.0000	35.0000	465.0000
97	6.0000	71.0000	743.0000
98	4.0000	57.0000	176.0000
99	7.0000	46.0000	898.0000
100	9.0000	71.0000	269.0000

Ожидаемый	0.5639	0.5862	0.5662
Полученный	0.6300	0.5700	0.6200

Рис. 3.2 – Пример работы для 100 чисел

```
~/Documents/Repositories/bmstu/MathModeling/sem_02/lab_01
> ./run.py 1000
```

Программный метод:

No	1 разр.	2 разр.	3 разр.
1	1.0000	95.0000	677.0000
2	3.0000	53.0000	654.0000
3	4.0000	26.0000	371.0000
4	4.0000	79.0000	959.0000
5	4.0000	58.0000	182.0000
...			
996	10.0000	17.0000	567.0000
997	10.0000	34.0000	158.0000
998	6.0000	34.0000	508.0000
999	8.0000	62.0000	329.0000
1000	9.0000	30.0000	985.0000

Ожидаемый	0.5789	0.5734	0.5875
Полученный	0.5940	0.5930	0.5700

Табличный метод:

No	1 разр.	2 разр.	3 разр.
1	7.0000	98.0000	615.0000
2	1.0000	56.0000	272.0000
3	7.0000	69.0000	643.0000
4	4.0000	99.0000	895.0000
5	9.0000	31.0000	117.0000
...			
996	4.0000	53.0000	213.0000
997	1.0000	55.0000	537.0000
998	3.0000	43.0000	248.0000
999	4.0000	85.0000	867.0000
1000	6.0000	51.0000	297.0000

Ожидаемый	0.5658	0.5895	0.5701
Полученный	0.6110	0.5750	0.5770

Рис. 3.3 – Пример работы для 1000 чисел

4 Вывод

Из результатов проведённой лабораторной работы справедливо сделать вывод, что, чем больше количество генерируемых случайных чисел, тем равномернее они распределены.

Кроме того, по полученным результатам видно, что программный метод создаёт более равномерную последовательность случайных чисел, чем табличный.