

МИНОБРНАУКИ РОССИИ  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ  
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ ИМЕНИ Н.Г. ЧЕРНЫШЕВСКОГО»**

Кафедра теоретических основ  
компьютерной безопасности и  
криптографии

**ТЕОРИЯ ПСЕВДОСЛУЧАЙНЫХ ГЕНЕРАТОРОВ**  
**ОТЧЁТ ПО ЛАБОРАТОРНОЙ РАБОТЕ**

студента 4 курса 431 группы  
специальности 10.05.01 Компьютерная безопасность  
факультета компьютерных наук и информационных технологий

Серебрякова Алексея Владимировича

Преподаватель  
Ассистент

\_\_\_\_\_ Н.А. Артемова

подпись, дата

Саратов 2023

# **1 Тестирование статистических свойств последовательности псевдослучайных чисел**

Описание задания:

Протестируйте статистические свойства последовательности псевдослучайных чисел:

1. Вычислить математическое ожидание последовательности;
2. Вычислить среднеквадратичное отклонение последовательности;
3. Сравните полученные оценки с заданными в пп. 1 параметрами. Постройте графики зависимостей оценок от объема выборки. Оцените относительные погрешности для какой-либо одной выборки.
4. Вычислить значение и дать ответ на вопрос удовлетворяет ли ППСЧ
  - a) Критерию хи-квадрат;
  - б) Критерию серий;
  - в) Критерию интервалов;
  - г) Критерию разбиений;
  - д) Критерию перестановок;
  - е) Критерию монотонности;
  - ж) Критерию конфликтов.

Описание используемых ПСЧ:

В работе рассматриваются последовательности псевдослучайных величин, сгенерированные в первой части практической работы следующими алгоритмами:

1. Линейный конгруэнтный метод;
2. Аддитивный метод;
3. Пятипараметрический метод;
4. Регистр сдвига с обратной связью (РСЛОС);
5. Нелинейная комбинация РСЛОС;
6. Вихрь Мерсенна;
7. RC4;
8. ГПСЧ на основе RSA;
9. Алгоритм Блюма-Блюма-Шуба.

После генерации данные последовательности с помощью программы из второй части практической работы были приведены к стандартному равномер-

ному распределению с входными параметрами 0 и 0,999, т.е. каждое число с из последовательности  $\in [0,999]$ .

### **1.1 Мат. ожидание, среднекв. отклонение, сравнение с теоретическими оценками и построение графиков**

Для того, чтобы найти **математическое ожидание** необходимо воспользоваться следующей формулой:

$$M = \frac{\sum_{i=1}^k x_i}{k}$$

где  $x_i$  - числа последовательности,  $k$  - количество таких чисел.

Для того, чтобы найти **среднеквадратичное отклонение** необходимо воспользоваться следующей формулой:

$$\sigma = \sqrt{\frac{\sum_{i=1}^k |x_i - M|^2}{k}}$$

где  $x_i$  - числа последовательности,  $k$  - количество таких чисел,  $M$  - математическое ожидание.

Теоретическая оценка при равномерном распределении для мат. ожидания равна 0.5, для среднеквадратичного отклонения примерно равна 0.2887.

Далее необходимо посчитать относительную погрешность для одной выборки. Для этого необходимо сравнить полученные результаты с эталонными и вычислить погрешности согласно следующим формулам:

$$\Delta M = \frac{|M - 0,5|}{M}, \Delta Q = \frac{|Q - \sqrt{1/12}|}{M}$$

Генератор	Мат.Ожидание	Среднекв.отклонение	Погрешность	
			Мат.Ожидания	Среднекв. отклонения
lc	0.4922	0.2894	0.0158	0.0024
add	0.4921	0.2887	0.0161	0.0010
5p	0.4950	0.2895	0.0101	0.0028
lfsr	0.4948	0.2902	0.0105	0.0052
nfsr	0.4988	0.2884	0.0024	0.0010
mt	0.5013	0.2878	0.0026	0.0031
rc4	0.5028	0.2903	0.0056	0.0055
rsa	0.4990	0.2876	0.0020	0.0038
bbs	0.5049	0.2891	0.0097	0.0014

Мат. ожидание последовательности: 0.4922. Её среднеквадратичное отклонение: 0.2894.  
 Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
 Относительные погрешности мат. ожидания: 0.0158; среднеквадратичного отклонения: 0.0024.  
 □

Рисунок 1 – Вычисление показателей для генератора lc

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2887.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0.

Рисунок 2 – Вычисление показателей для генератора add

Мат. ожидание последовательности: 0.495. Её среднеквадратичное отклонение: 0.2895.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0101; среднеквадратичного отклонения: 0.0028.

Рисунок 3 – Вычисление показателей для генератора 5р

Мат. ожидание последовательности: 0.4948. Её среднеквадратичное отклонение: 0.2902.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0105; среднеквадратичного отклонения: 0.0052.

Рисунок 4 – Вычисление показателей для генератора lfsr

Мат. ожидание последовательности: 0.4988. Её среднеквадратичное отклонение: 0.2884.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0024; среднеквадратичного отклонения: 0.001.

Рисунок 5 – Вычисление показателей для генератора nfsr

Мат. ожидание последовательности: 0.5013. Её среднеквадратичное отклонение: 0.2878.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0026; среднеквадратичного отклонения: 0.0031.

Рисунок 6 – Вычисление показателей для генератора mt

Мат. ожидание последовательности: 0.5028. Её среднеквадратичное отклонение: 0.2903.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0056; среднеквадратичного отклонения: 0.0055.

Рисунок 7 – Вычисление показателей для генератора rc4

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2876.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.0038.

Рисунок 8 – Вычисление показателей для генератора rsa

Мат. ожидание последовательности: 0.5049. Её среднеквадратичное отклонение: 0.2891.  
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.0014.

Рисунок 9 – Вычисление показателей для генератора bbs

Далее построим графики зависимости мат. ожидания от объема выборки и среднеквадратичного отклонения от объема выборки. Шаг: 50.

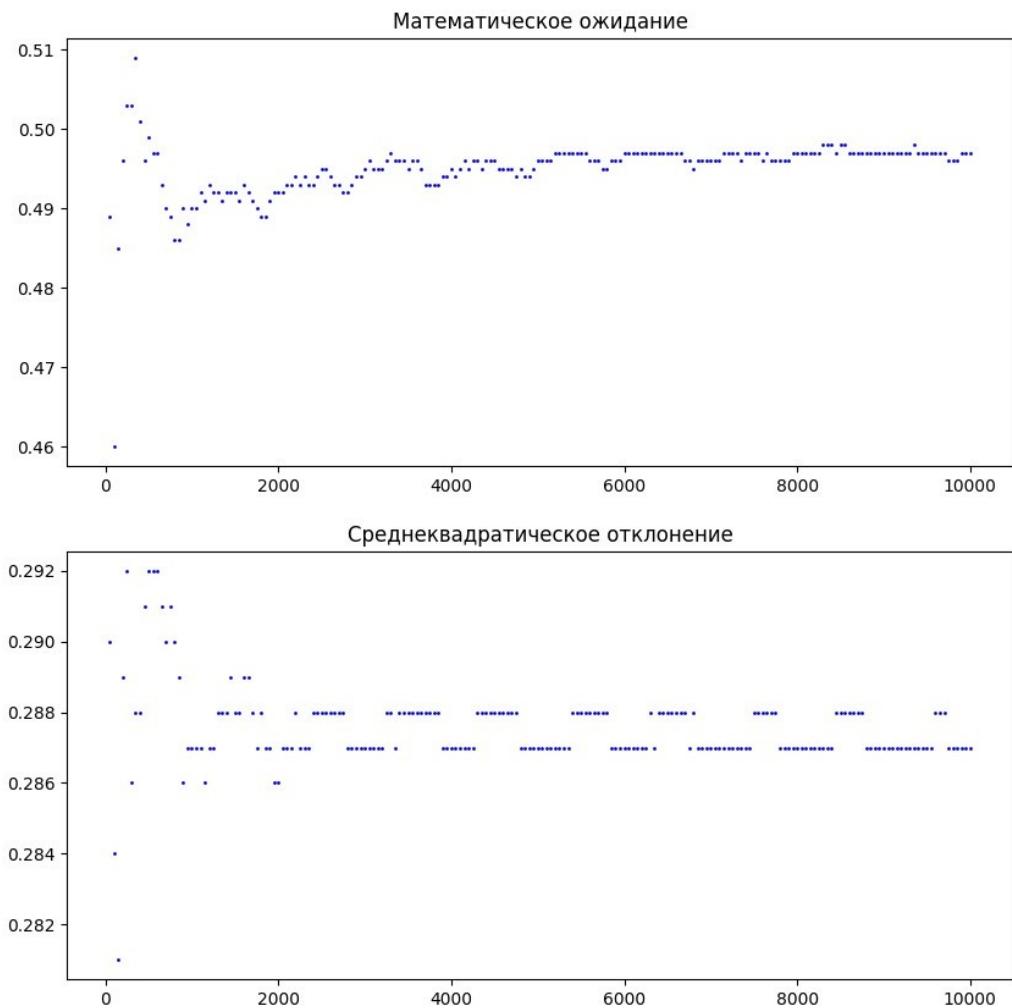


Рисунок 10 – Графики зависимости для генератора lc

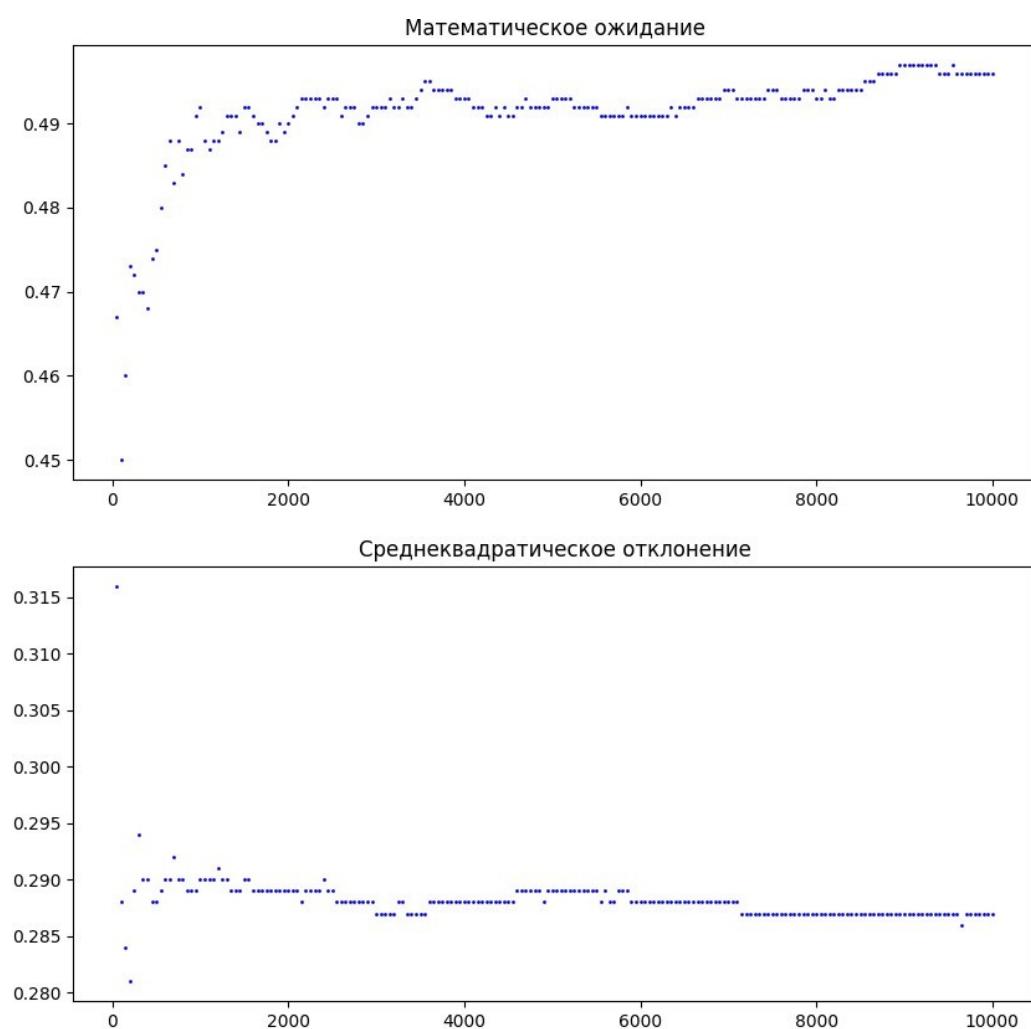


Рисунок 11 – Графики зависимости для генератора add

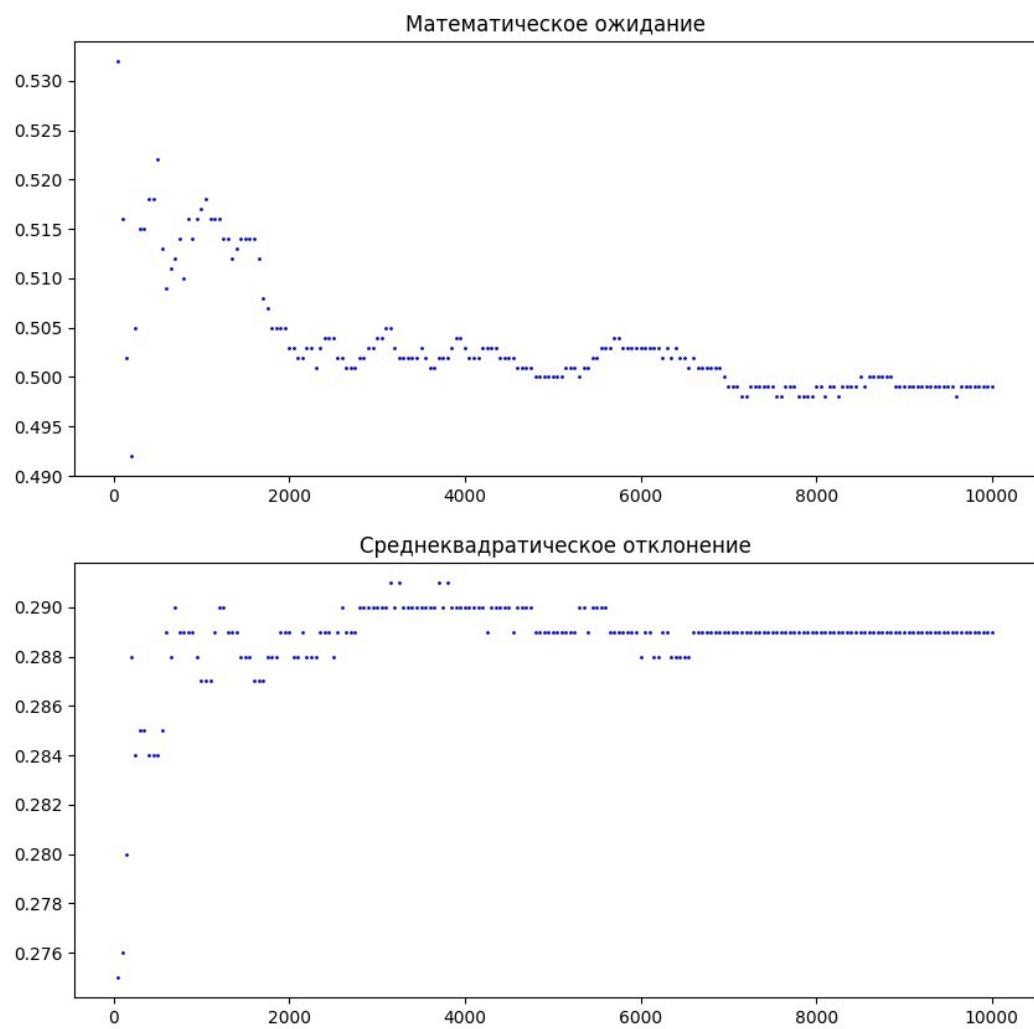


Рисунок 12 – Графики зависимости для генератора 5р

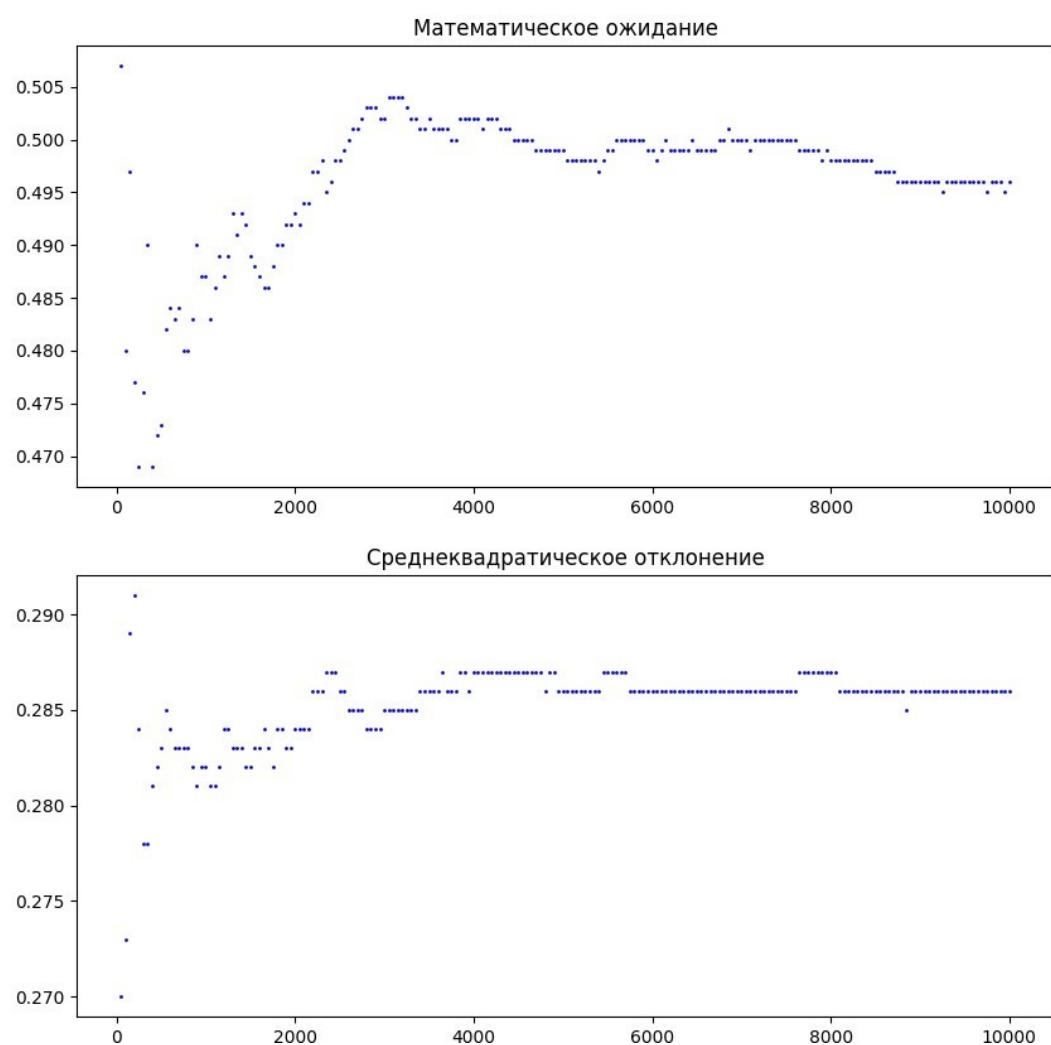


Рисунок 13 – Графики зависимости для генератора lfsr

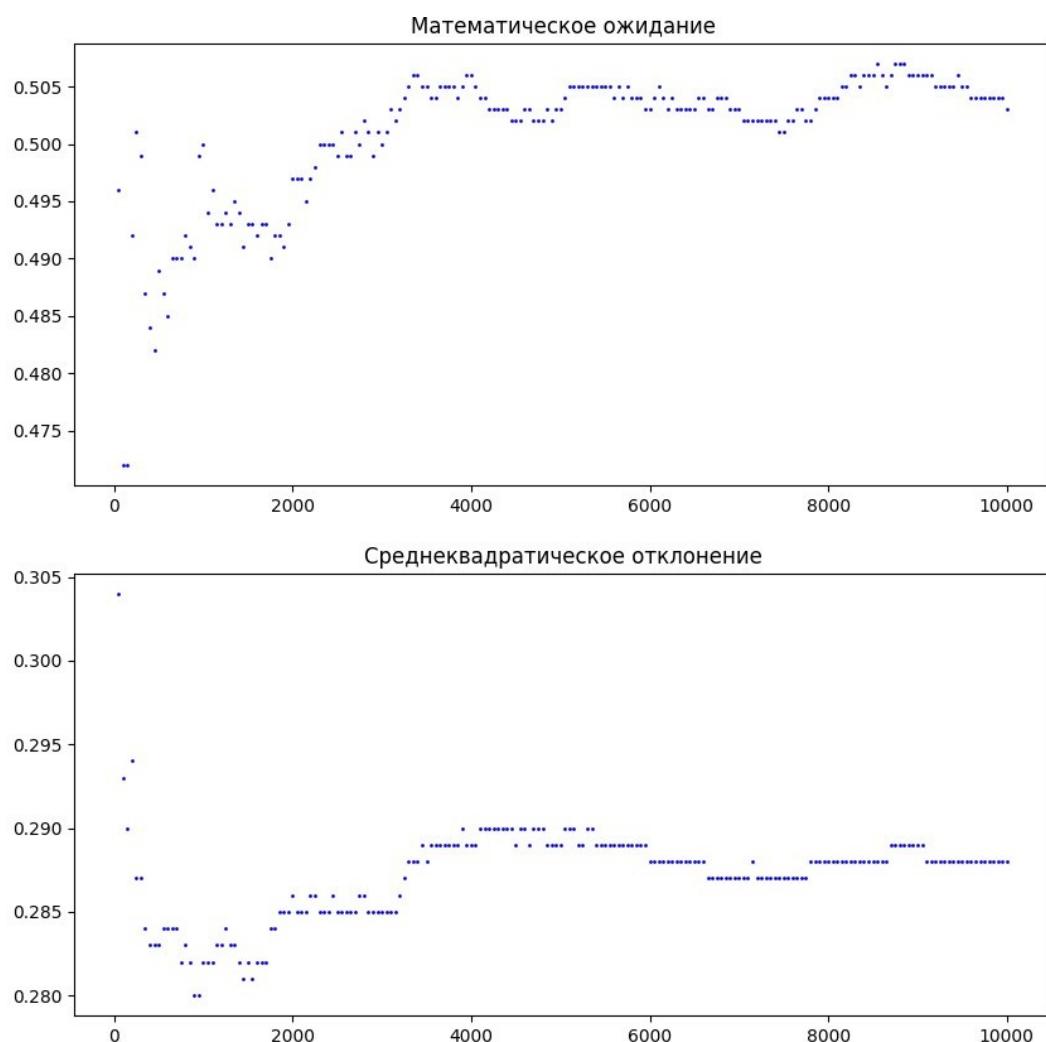


Рисунок 14 – Графики зависимости для генератора nfsr

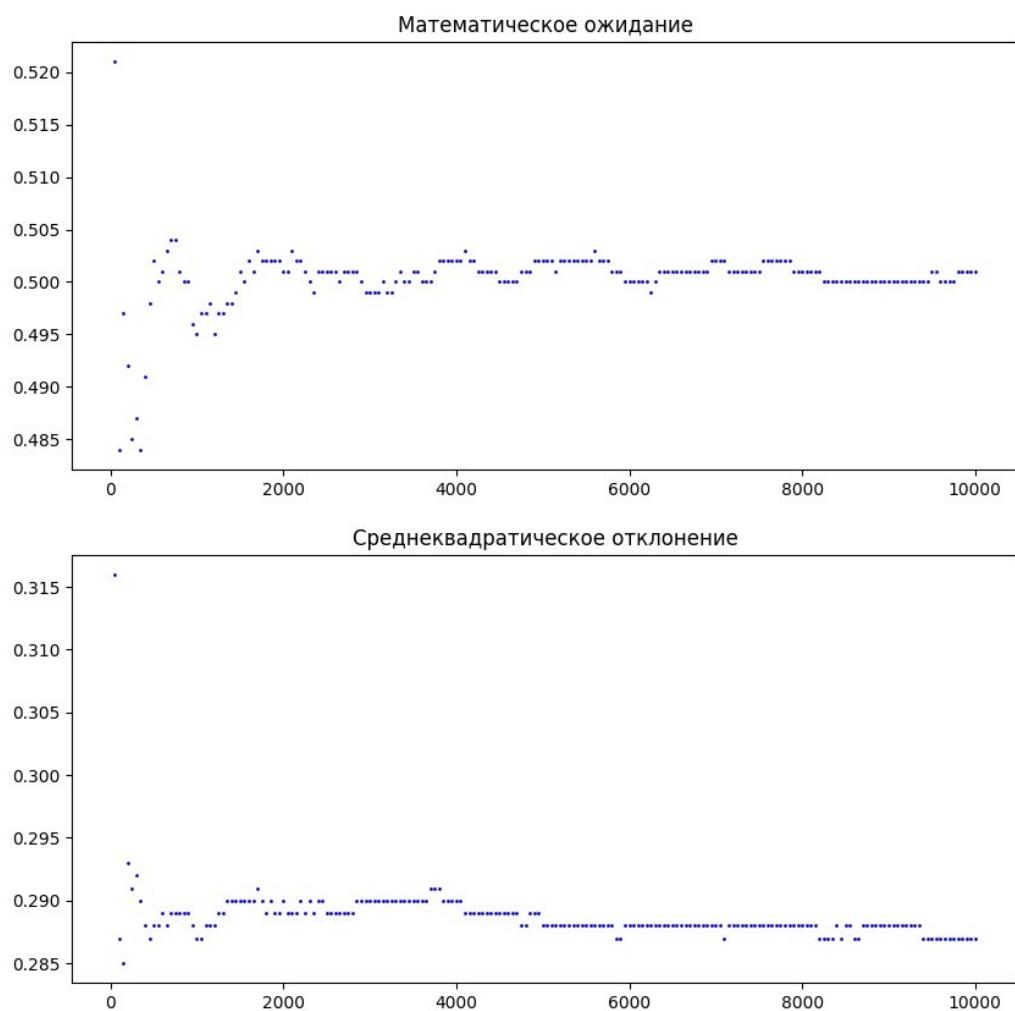


Рисунок 15 – Графики зависимости для генератора mt

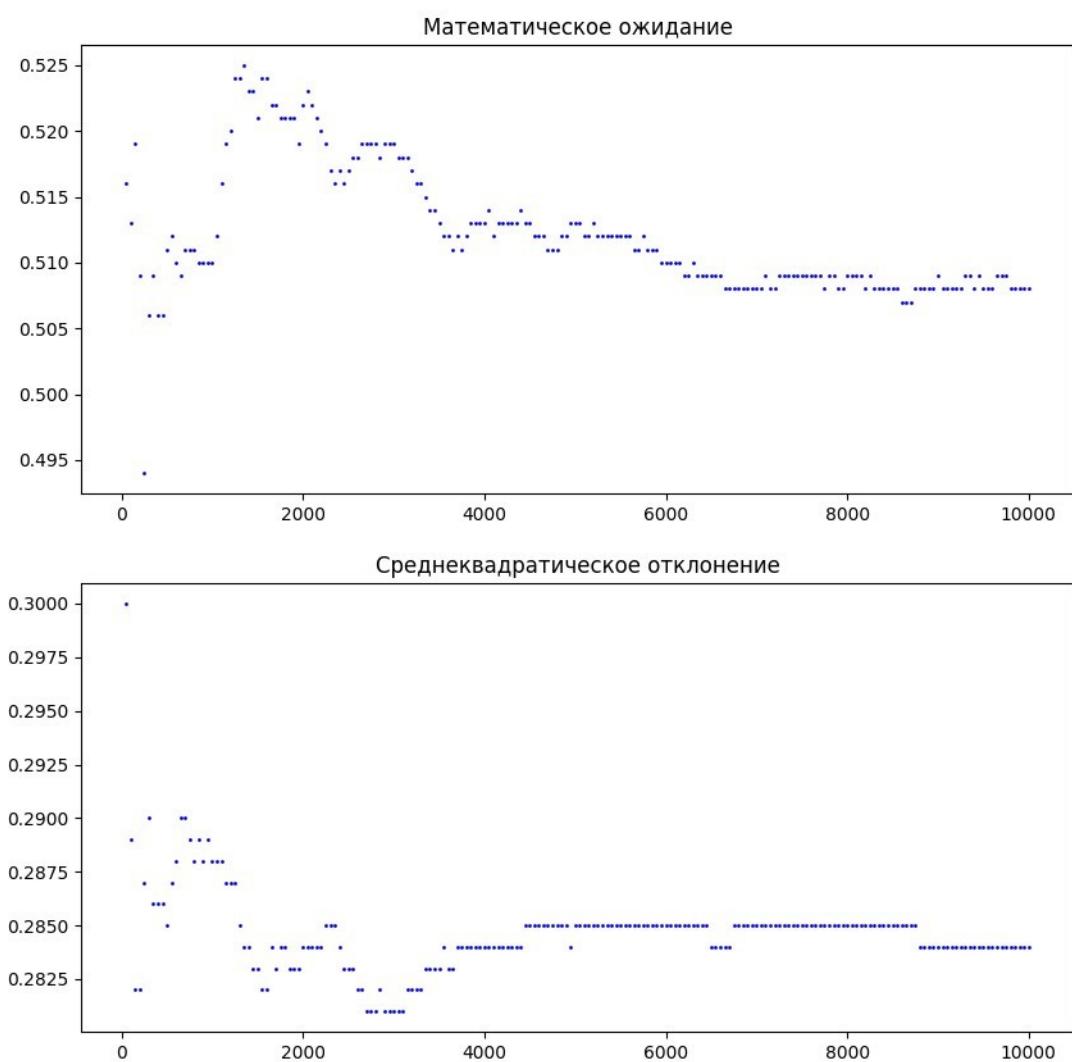


Рисунок 16 – Графики зависимости для генератора rc4

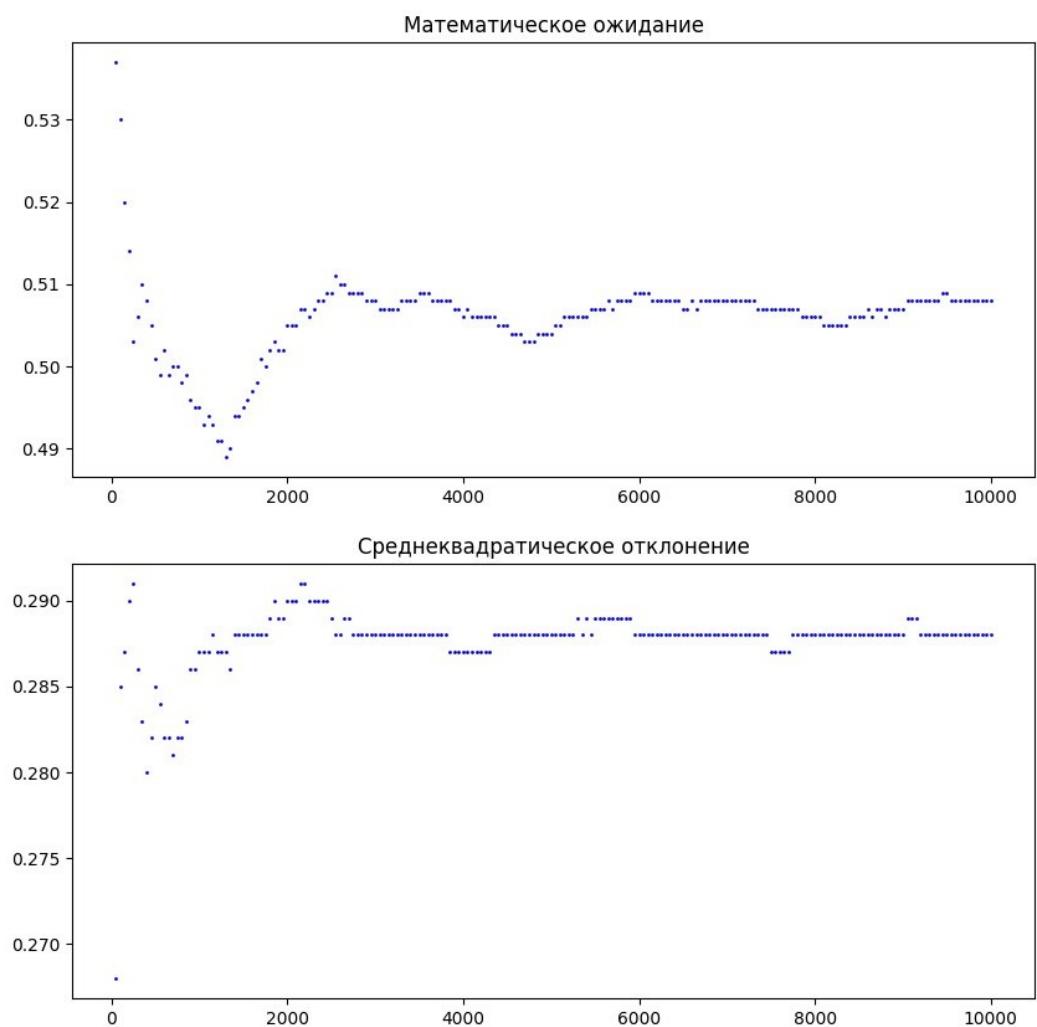


Рисунок 17 – Графики зависимости для генератора rsa

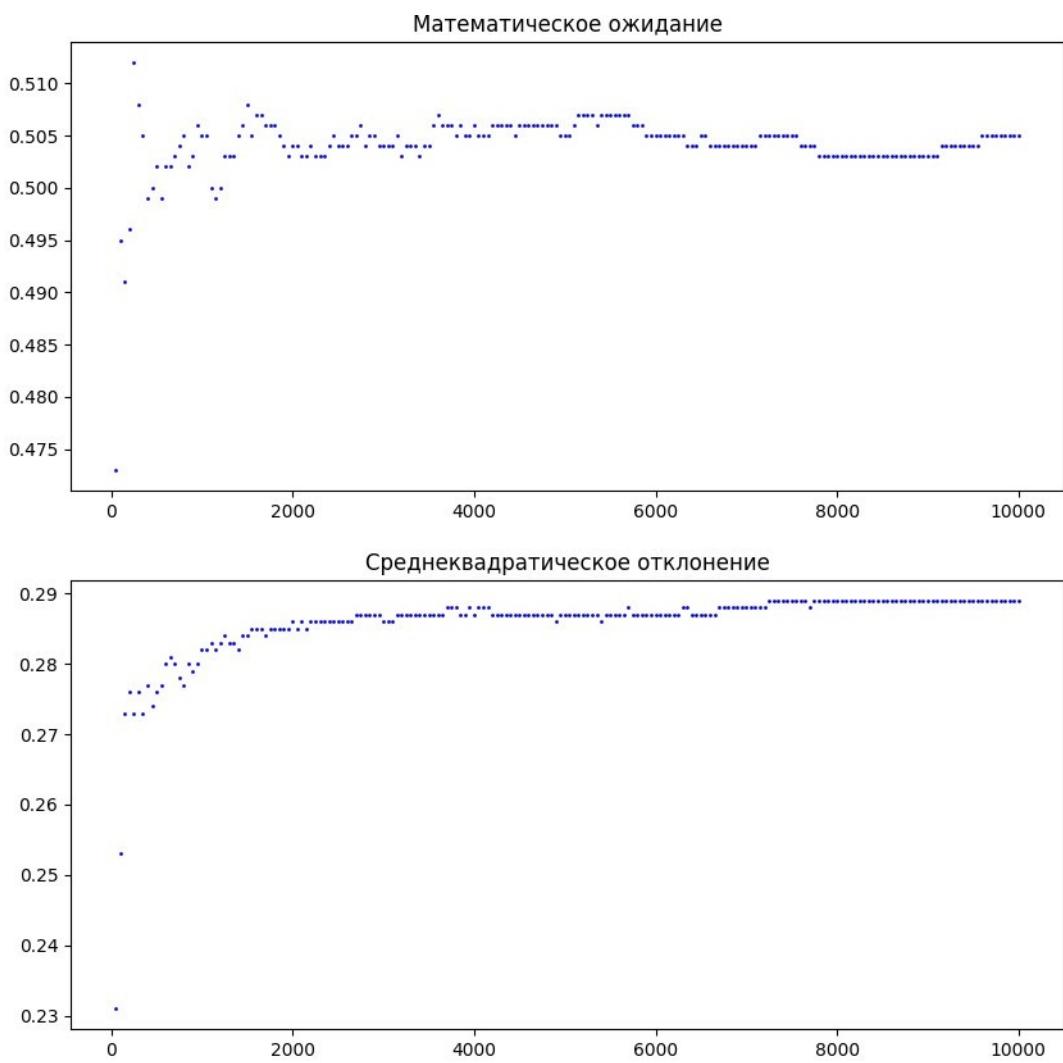


Рисунок 18 – Графики зависимости для генератора bbs

## 1.2 Сравнение полученных оценок с теоретическими и построение графиков

### 1.3 Проверка критериев

#### 1.3.1 Критерий хи-квадрат

Описание критерия:

Проверка критерия  $\chi^2$  для некоторой последовательности чисел (или наблюдений величины X) будет состоять из следующих шагов:

1. Выполняем достаточное число независимых наблюдений.
2. Подсчитываем число  $n_i$  наблюдений попавших в каждый из интервалов  $(a_i, b_i]$ ,  $i = 1, \dots, k$ .
3. Подсчитываем статистику

$$\chi^2 = \sum_{j=1}^k \frac{(n_j - E_j)^2}{E_j}$$

4. Определяем, находится ли вычисленная в доверительном интервале.

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [762, 733, 732, 715, 735, 727, 704, 739, 704, 730, 655, 64
7, 683, 734].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 19.6152.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.4922. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0158; среднеквадратичного отклонения: 0.0028.
```

Рисунок 19 – Результаты выполнения программы для генератора lc

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_add /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [749, 753, 735, 745, 703, 669, 733, 702, 710, 732, 697, 69
6, 723, 653].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 16.174.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2887.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0.
```

Рисунок 20 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [743, 757, 732, 697, 734, 699, 694, 693, 709, 722, 724, 70
5, 725, 666].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 10.448.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.4951. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0028.
```

Рисунок 21 – Результаты выполнения программы для генератора 5p

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [751, 785, 704, 717, 666, 694, 687, 785, 705, 692, 718, 69
0, 679, 727].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 24.56.
Результат о принятии гипотезы: не принята.

Мат. ожидание последовательности: 0.495. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0101; среднеквадратичного отклонения: 0.0052.
```

Рисунок 22 – Результаты выполнения программы для генератора lfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [716, 721, 711, 746, 673, 717, 708, 732, 732, 712, 708, 70
2, 714, 708].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 5.1504.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.4989. Её среднеквадратичное отклонение: 0.2884.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.001.
```

Рисунок 23 – Результаты выполнения программы для генератора nfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [691, 707, 729, 719, 738, 698, 687, 727, 713, 748, 709, 71
5, 696, 723].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 5.8028.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.5011. Её среднеквадратичное отклонение: 0.2878.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.0031.
```

Рисунок 24 – Результаты выполнения программы для генератора mt

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:a
Используемые диапазоны интервалов: [(0.0, 0.0711), (0.0711, 0.1422), (0.1422, 0.2133), (0.2133, 0.2844),
(0.2844, 0.3555), (0.3555, 0.4266), (0.4266, 0.4977), (0.4977, 0.5688), (0.5688, 0.6399), (0.6399, 0.711),
(0.711, 0.7821), (0.7821, 0.8532), (0.8532, 0.9243), (0.9243, 0.9954)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [746, 710, 691, 680, 692, 685, 667, 683, 761, 768, 730, 66
0, 761, 718].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 24.8756.
Результат о принятии гипотезы: не принята.

Мат. ожидание последовательности: 0.503. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.006; среднеквадратичного отклонения: 0.0052.
```

Рисунок 25 – Результаты выполнения программы для генератора rc4

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [703, 733, 734, 682, 714, 695, 727, 732, 719, 719, 713, 75
7, 676, 696].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 8.9976.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.4991. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0018; среднеквадратичного отклонения: 0.0038.
```

Рисунок 26 – Результаты выполнения программы для генератора rsa

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:a
Используемые диапазоны интервалов: [(0.0, 0.0714), (0.0714, 0.1428), (0.1428, 0.2142), (0.2142, 0.2856),
(0.2856, 0.357), (0.357, 0.4284), (0.4284, 0.4998), (0.4998, 0.5712), (0.5712, 0.6426), (0.6426, 0.714),
(0.714, 0.7854), (0.7854, 0.8568), (0.8568, 0.9282), (0.9282, 0.9996)]
Количество степеней свободы: 13. Уровень значимости: 0.05. Критическое значение хи-квадрат: 22.362.
Наблюдаемое распределение чисел по интервалам: [713, 720, 673, 716, 703, 663, 727, 731, 672, 708, 777, 74
8, 740, 709].
Ожидаемое распределение чисел по интервалам: 714.286.
Значение хи-квадрат: 17.5376.
Результат о принятии гипотезы: принята.

Мат. ожидание последовательности: 0.505. Её среднеквадратичное отклонение: 0.2891.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0014.
```

Рисунок 27 – Результаты выполнения программы для генератора bbs

### 1.3.2 Критерий серий

Описание критерия:

Критерий серий позволяет убедиться в том, что пары последовательных чисел равномерно распределены независимым образом. Проверка критерия проводится следующим образом:

1. Воспользуемся критерием Хи-квадрат. Для этого преобразуем последовательность  $X^n = (x_1, x_2, \dots, x_n)$  в  $Y^n = (d[x_1], d[x_2], \dots, d[x_n])$  с некоторым  $d$  (в реализации  $d = 4$ ).
2. Подсчитываем количество совпадений

$$\left( y_{2j}, y_{2j+1} \right) = (q, r), 0 \leq j \leq n, 0 \leq q, 0 \leq d$$

3. Применяем Хи-квадрат к полученному набору с параметрами:

$$k = d^2, p_j = \frac{1}{d^2}$$

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:b
Совпадения пар последовательных чисел: {(0, 0): 344, (0, 1): 336, (0, 2): 300, (0, 3): 312, (1, 0): 302,
(1, 1): 323, (1, 2): 303, (1, 3): 281, (2, 0): 342, (2, 1): 341, (2, 2): 304, (2, 3): 313, (3, 0): 302, (3, 1):
320, (3, 2): 303, (3, 3): 274}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 20.794.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2894.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0024.
```

Рисунок 28 – Результаты выполнения программы для генератора lc

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_add /c:b
Совпадения пар последовательных чисел: {(0, 0): 338, (0, 1): 324, (0, 2): 355, (0, 3): 309, (1, 0): 345,
(1, 1): 307, (1, 2): 318, (1, 3): 278, (2, 0): 289, (2, 1): 294, (2, 2): 320, (2, 3): 285, (3, 0): 313, (3, 1): 305, (3, 2): 324, (3, 3): 296}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 22.643.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.492. Её среднеквадратичное отклонение: 0.2888.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0163; среднеквадратичного отклонения: 0.0003.
```

Рисунок 29 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:b
Совпадения пар последовательных чисел: {(0, 0): 343, (0, 1): 334, (0, 2): 320, (0, 3): 316, (1, 0): 328,
(1, 1): 301, (1, 2): 305, (1, 3): 300, (2, 0): 298, (2, 1): 303, (2, 2): 304, (2, 3): 333, (3, 0): 278, (3, 1): 325, (3, 2): 289, (3, 3): 323}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 15.514.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.4951. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0028.
```

Рисунок 30 – Результаты выполнения программы для генератора 5p

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:b
Совпадения пар последовательных чисел: {(0, 0): 351, (0, 1): 306, (0, 2): 292, (0, 3): 313, (1, 0): 316,
(1, 1): 319, (1, 2): 316, (1, 3): 266, (2, 0): 343, (2, 1): 280, (2, 2): 327, (2, 3): 308, (3, 0): 324, (3, 1): 286, (3, 2): 342, (3, 3): 311}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 25.914.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.495. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0101; среднеквадратичного отклонения: 0.0052.
```

Рисунок 31 – Результаты выполнения программы для генератора lfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:b
Совпадения пар последовательных чисел: {(0, 0): 326, (0, 1): 308, (0, 2): 308, (0, 3): 293, (1, 0): 301,
(1, 1): 315, (1, 2): 315, (1, 3): 302, (2, 0): 328, (2, 1): 307, (2, 2): 302, (2, 3): 321, (3, 0): 337, (3, 1): 304, (3, 2): 327, (3, 3): 306}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 7.155.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.4989. Её среднеквадратичное отклонение: 0.2884.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.001.
```

Рисунок 32 – Результаты выполнения программы для генератора nfsr

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:b
Совпадения пар последовательных чисел: {(0, 0): 302, (0, 1): 341, (0, 2): 278, (0, 3): 326, (1, 0): 299,
(1, 1): 304, (1, 2): 291, (1, 3): 306, (2, 0): 314, (2, 1): 352, (2, 2): 338, (2, 3): 313, (3, 0): 303, (
3, 1): 308, (3, 2): 327, (3, 3): 298}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 18.554.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.5011. Её среднеквадратичное отклонение: 0.2878.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.0031.

```

Рисунок 33 – Результаты выполнения программы для генератора mt

```

[alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:b
Совпадения пар последовательных чисел: {(0, 0): 305, (0, 1): 290, (0, 2): 313, (0, 3): 304, (1, 0): 304,
(1, 1): 311, (1, 2): 297, (1, 3): 298, (2, 0): 317, (2, 1): 322, (2, 2): 334, (2, 3): 336, (3, 0): 356, (
3, 1): 289, (3, 2): 305, (3, 3): 319}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 15.45.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.5029. Её среднеквадратичное отклонение: 0.2904.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0058; среднеквадратичного отклонения: 0.0059.

```

Рисунок 34 – Результаты выполнения программы для генератора rc4

```

[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:b
Совпадения пар последовательных чисел: {(0, 0): 333, (0, 1): 298, (0, 2): 307, (0, 3): 315, (1, 0): 307,
(1, 1): 317, (1, 2): 323, (1, 3): 307, (2, 0): 292, (2, 1): 324, (2, 2): 329, (2, 3): 311, (3, 0): 302, (
3, 1): 311, (3, 2): 303, (3, 3): 321}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 6.272.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.4989. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.0038.

```

Рисунок 35 – Результаты выполнения программы для генератора rsa

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:b
Совпадения пар последовательных чисел: {(0, 0): 285, (0, 1): 322, (0, 2): 311, (0, 3): 330, (1, 0): 302,
(1, 1): 282, (1, 2): 293, (1, 3): 316, (2, 0): 294, (2, 1): 325, (2, 2): 318, (2, 3): 327, (3, 0): 338, (
3, 1): 327, (3, 2): 296, (3, 3): 334}.
Теоретическое количество попаданий в каждую категорию: 312.5.
Уровень значимости: 0.05.
Параметр d принимает значение: 4.
Критическое значение для критерия серий с 16 степенями свободы: 26.296.
Вычисленное значение хи-квадрат: 15.75.
Представленная последовательность удовлетворяет критерию серий: True.

Мат. ожидание последовательности: 0.5049. Её среднеквадратичное отклонение: 0.289.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.001.

```

Рисунок 36 – Результаты выполнения программы для генератора bbs

### 1.3.3 Критерий интервалов

Описание критерия:

Пусть  $a$  и  $b$  – два действительных числа таких, что  $0 \leq a \leq b$ . Рассмотрим

рим длины подпоследовательностей  $x_j, x_{j+1}, \dots, x_{j+r}$ , в которых  $x_j, x_{j+1}, \dots, x_{j+r-1} \notin [a, b]$ ,  $x_{j+r} \in [a, b]$ . Такую последовательность будем называть интервалом длины  $r$ .

Сначала, нам нужно подсчитать число интервалов длиной  $0, 1, \dots, n$ .

Шаги алгоритма подсчета числа интервалов:

1. Инициализация. Присвоить  $j = -1, s = 0, c_r = 0, 0 \leq r \leq t$ .
2.  $r = 0$ .
3.  $j = j + 1$ . Если  $a \leq x_j \leq b$ , то переход на шаг 5.
4.  $r = r + 1$ . Переход к шагу 3.
5. Если  $r \leq t$ , то  $c_t = c_t + 1$ , иначе  $-c_r = c_r + 1$ .
6.  $s = s + 1$ . Если  $s < n$  то переход на шаг 2.

После этого мы можем применить хи-квадрат критерий для  $k = t + 1$  к значениям  $c_i, i = 0, 1, \dots, t$  с параметрами

$$p_r = p(1-p)r \text{ для } 0 \leq r \leq t - 1; p_t = (1-p)t; p = (a - b).$$

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 25.865.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1995.

Используемые значения границ alpha = 0.397, beta = 0.663.
Представленная последовательность удовлетворяет критерию интервалов: False.
Пересчитанные вероятности p_r и p_t: [0.266, 0.195, 0.143, 0.105, 0.077, 0.057, 0.042, 0.031, 0.0842].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 478, 1: 383, 2: 287, 3: 227, 4: 166,
5: 125, 6: 61, 7: 58, 8: 210}.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0028.
```

Рисунок 37 – Результаты выполнения программы для генератора lc

```
o [alse0722@gavno lab3]$ python3 rnt.py /f:st_add /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 10.334.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1960.

Используемые значения границ alpha = 0.373, beta = 0.698.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_r и p_t: [0.325, 0.219, 0.148, 0.1, 0.067, 0.046, 0.031, 0.021, 0.0431].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 624, 1: 436, 2: 300, 3: 181, 4: 141,
5: 77, 6: 63, 7: 34, 8: 104}.

Мат. ожидание последовательности: 0.4922. Её среднеквадратичное отклонение: 0.2887.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0158; среднеквадратичного отклонения: 0.0.
```

Рисунок 38 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 8.243.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1066
.
Используемые значения границ alpha = 0.258, beta = 0.536.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_g и p_t: [0.278, 0.201, 0.145, 0.105, 0.076, 0.055, 0.039, 0.028, 0.0738].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 274, 1: 237, 2: 140, 3: 112, 4: 78,
5: 58, 6: 51, 7: 32, 8: 84}.

Мат. ожидание последовательности: 0.4951. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0028.
```

Рисунок 39 – Результаты выполнения программы для генератора 5p

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 4.67.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1565
.
Используемые значения границ alpha = 0.46, beta = 0.825.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_g и p_t: [0.365, 0.232, 0.147, 0.093, 0.059, 0.038, 0.024, 0.015, 0.0264].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 597, 1: 344, 2: 226, 3: 148, 4: 94,
5: 56, 6: 35, 7: 29, 8: 36}.

Мат. ожидание последовательности: 0.4949. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0103; среднеквадратичного отклонения: 0.0052.
```

Рисунок 40 – Результаты выполнения программы для генератора lfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 6.48.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1958
.
Используемые значения границ alpha = 0.087, beta = 0.468.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_g и p_t: [0.381, 0.236, 0.146, 0.09, 0.056, 0.035, 0.021, 0.013, 0.0216].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 764, 1: 456, 2: 307, 3: 153, 4: 108,
5: 60, 6: 41, 7: 24, 8: 45}.

Мат. ожидание последовательности: 0.4989. Её среднеквадратичное отклонение: 0.2885.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.0007.
```

Рисунок 41 – Результаты выполнения программы для генератора nfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 4.653.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1362
.
Используемые значения границ alpha = 0.329, beta = 0.688.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_g и p_t: [0.359, 0.23, 0.148, 0.095, 0.061, 0.039, 0.025, 0.016, 0.0285].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 490, 1: 297, 2: 202, 3: 129, 4: 90,
5: 61, 6: 40, 7: 20, 8: 33}.

Мат. ожидание последовательности: 0.5011. Её среднеквадратичное отклонение: 0.2878.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.0031.
```

Рисунок 42 – Результаты выполнения программы для генератора mt

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 0.764.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1670
.
Используемые значения границ alpha = 0.076, beta = 0.288.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_r и p_t: [0.212, 0.167, 0.132, 0.104, 0.082, 0.064, 0.051, 0.04, 0.1487].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 346, 1: 280, 2: 218, 3: 178, 4: 138,
5: 103, 6: 85, 7: 71, 8: 251}.

Мат. ожидание последовательности: 0.503. Её среднеквадратичное отклонение: 0.2901.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.006; среднеквадратичного отклонения: 0.0048.
```

Рисунок 43 – Результаты выполнения программы для генератора rc4

```
^[[A[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 5.293.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1204
.
Используемые значения границ alpha = 0.492, beta = 0.768.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_r и p_t: [0.276, 0.2, 0.145, 0.105, 0.076, 0.055, 0.04, 0.029, 0.0755].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 315, 1: 255, 2: 183, 3: 120, 4: 103,
5: 62, 6: 49, 7: 29, 8: 88}.

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.0038.
```

Рисунок 44 – Результаты выполнения программы для генератора rsa

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:c
Уровень значимости: 0.05.
Критическое значение для критерия хи-квадрат с 9 степенями свободы: 16.919.
Найденное оптимальное значение хи-квадрата: 2.732.
Это значение было найдено при параметрах t (макс. длина интервала) = 8 и n (количество интервалов) = 1449
.
Используемые значения границ alpha = 0.192, beta = 0.474.
Представленная последовательность удовлетворяет критерию интервалов: True.
Пересчитанные вероятности p_r и p_t: [0.282, 0.202, 0.145, 0.104, 0.075, 0.054, 0.039, 0.028, 0.0706].
Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {0: 394, 1: 291, 2: 218, 3: 146, 4: 108,
5: 76, 6: 63, 7: 46, 8: 107}.

Мат. ожидание последовательности: 0.505. Её среднеквадратичное отклонение: 0.2892.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0017.
```

Рисунок 45 – Результаты выполнения программы для генератора bbs

### 1.3.4 Критерий разбиений

Описание критерия:

В общем случае критерия разбиений рассматриваются  $n$  групп  $k$  последовательных чисел, и подсчитывается число групп из  $k$  чисел с  $r$  различными числами. Затем применяется хи-квадрат критерий, в котором используются вероятности того, что в группе  $r$  различных чисел

$$p_r = \frac{d(d-1)\dots(d-r+1)}{d^k} \binom{k}{r}$$

Здесь  $\binom{k}{r} = S(n,k)$  – числа Стирлинга, задающие число способов разбиения множества из  $n$  элементов на  $k$  непересекающихся подмножеств, которые можно вычислить по формуле:

$$S(n,k) = \frac{1}{k!} \sum_{j=0}^k (-1)^{k+j} \binom{k}{j} j^n$$

Так как вероятности  $p_r$  очень малы, когда  $r = 1$  или  $2$ , следует, перед применением критерия хи-квадрат, объединить несколько категорий, имеющих малые вероятности в одну. Чтобы получить формулу для  $p_r$ , следует подсчитать, сколько  $d^k$  групп из  $k$  чисел, расположенных между  $0$  и  $d - 1$ , имеют точно  $r$  различных элементов, и разделить это число на  $d^k$ .

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 430, 4: 1025, 3: 477, 2: 68, 1: 0}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 9.341.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.4922. Её среднеквадратичное отклонение: 0.2894.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0158; среднеквадратичного отклонения: 0.0024.
```

Рисунок 46 – Результаты выполнения программы для генератора lc

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_add /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 417, 4: 1019, 3: 511, 2: 51, 1: 2}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 6.471.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2887.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0.
```

Рисунок 47 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 438, 4: 1026, 3: 487, 2: 48, 1: 1}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 4.239.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.4951. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0028.
```

Рисунок 48 – Результаты выполнения программы для генератора 5p

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 388, 4: 1056, 3: 504, 2: 51, 1: 1}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 3.134.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.4948. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0105; среднеквадратичного отклонения: 0.0052.
```

Рисунок 49 – Результаты выполнения программы для генератора lfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 421, 4: 1002, 3: 525, 2: 51, 1: 1}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 1.994.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2884.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.001.
```

Рисунок 50 – Результаты выполнения программы для генератора nfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 409, 4: 1026, 3: 515, 2: 50, 1: 0}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 0.4457.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.5013. Её среднеквадратичное отклонение: 0.2878.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0026; среднеквадратичного отклонения: 0.0031.
```

Рисунок 51 – Результаты выполнения программы для генератора mt

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 402, 4: 1001, 3: 540, 2: 57, 1: 0}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 3.268.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.5028. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0056; среднеквадратичного отклонения: 0.0052.
```

Рисунок 52 – Результаты выполнения программы для генератора rc4

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): { 5: 444, 4: 1003, 3: 496, 2: 57, 1: 0}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 4.871.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.0038.
```

Рисунок 53 – Результаты выполнения программы для генератора rsa

```

○ [false0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:d
Уровень значимости: 0.05.
Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в группе): {5: 386, 4: 1047, 3: 523, 2: 43, 1: 1}.
Вычисленные теоретические вероятности частоты комбинаций: [0.2051, 0.5127, 0.2563, 0.0256, 0.0002].
Критическое значение для критерия хи-квадрат с 4 степенями свободы: 9.488.
Найденное значение хи-квадрат: 4.282.
Представленная последовательность удовлетворяет критерию разбиений: True.

Мат. ожидание последовательности: 0.5049. Её среднеквадратичное отклонение: 0.2891.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.0014.

```

Рисунок 54 – Результаты выполнения программы для генератора bbs

### 1.3.5 Критерий перестановок

Описание критерия:

Последовательность  $X^m = (x_1, x_2, \dots, x_m)$  разбивается на  $n$  групп по  $t$

элементов в каждой:

$$u_j = (x_{jt}, x_{jt+1}, \dots, x_{jt+t-1}), 0 \leq h \leq n$$

Элементы в каждой группе можно упорядочивать  $t!$  различными способами. Подсчитывается число групп с любым возможным порядком и применяется хи-квадрат критерий с  $k = t!$  возможными категориями и вероятностью  $\frac{1}{t!}$  для каждой категории. В этом критерии предполагается, что  $x_s$  не могут быть равны между собой. В реализации берется  $t = 4$ .

Приведем результаты выполнения программы для каждого из генераторов:

```

○ [false0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 88, 1: 112, 2: 80, 3: 130, 4: 74, 5: 106, 6: 72, 7: 116, 8: 1
64, 9: 65, 10: 121, 11: 82, 12: 150, 13: 86, 14: 98, 15: 107, 16: 96, 17: 114, 18: 86, 19: 110, 20: 65, 2
1: 94, 22: 107, 23: 115}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 137.505.
Представленная последовательность удовлетворяет критерию перестановок: False.

Мат. ожидание последовательности: 0.4923. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0156; среднеквадратичного отклонения: 0.0028.

```

Рисунок 55 – Результаты выполнения программы для генератора lc

```

○ [false0722@gavno lab3]$ python3 rnt.py /f:st_add /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 109, 1: 115, 2: 113, 3: 114, 4: 95, 5: 96, 6: 98, 7: 92, 8: 9
0, 9: 124, 10: 107, 11: 90, 12: 102, 13: 108, 14: 108, 15: 110, 16: 109, 17: 101, 18: 95, 19: 97, 20: 110
, 21: 106, 22: 92, 23: 99}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 18.61.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2887.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0.

```

Рисунок 56 – Результаты выполнения программы для генератора add

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 99, 1: 130, 2: 101, 3: 107, 4: 96, 5: 102, 6: 87, 7: 122, 8: 108, 9: 113, 10: 107, 11: 100, 12: 90, 13: 94, 14: 104, 15: 97, 16: 102, 17: 108, 18: 100, 19: 105, 20: 101, 21: 97, 22: 118, 23: 97}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 21.457.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.4952. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.0028.

```

Рисунок 57 – Результаты выполнения программы для генератора 5p

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 123, 1: 103, 2: 109, 3: 130, 4: 100, 5: 105, 6: 110, 7: 111, 8: 101, 9: 79, 10: 101, 11: 110, 12: 106, 13: 101, 14: 92, 15: 100, 16: 107, 17: 116, 18: 98, 19: 105, 20: 92, 21: 85, 22: 87, 23: 115}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 30.152.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.495. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0101; среднеквадратичного отклонения: 0.0052.

```

Рисунок 58 – Результаты выполнения программы для генератора lfsr

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 100, 1: 106, 2: 109, 3: 94, 4: 94, 5: 111, 6: 92, 7: 107, 8: 96, 9: 85, 10: 108, 11: 120, 12: 117, 13: 103, 14: 109, 15: 112, 16: 100, 17: 110, 18: 107, 19: 88, 20: 98, 21: 110, 22: 107, 23: 104}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 17.552.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.4989. Её среднеквадратичное отклонение: 0.2884.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0022; среднеквадратичного отклонения: 0.001.

```

Рисунок 59 – Результаты выполнения программы для генератора nfsr

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 104, 1: 110, 2: 88, 3: 89, 4: 119, 5: 90, 6: 119, 7: 99, 8: 101, 9: 88, 10: 115, 11: 93, 12: 109, 13: 116, 14: 96, 15: 115, 16: 95, 17: 102, 18: 100, 19: 99, 20: 102, 21: 105, 22: 115, 23: 109}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 22.724.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.5013. Её среднеквадратичное отклонение: 0.2879.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0026; среднеквадратичного отклонения: 0.0028.

```

Рисунок 60 – Результаты выполнения программы для генератора mt

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 117, 1: 108, 2: 81, 3: 91, 4: 102, 5: 111, 6: 105, 7: 125, 8: 108, 9: 91, 10: 108, 11: 97, 12: 101, 13: 107, 14: 105, 15: 99, 16: 99, 17: 109, 18: 101, 19: 85, 20: 85, 21: 90, 22: 108, 23: 110}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 26.962.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.5027. Её среднеквадратичное отклонение: 0.2904.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0054; среднеквадратичного отклонения: 0.0059.

```

Рисунок 61 – Результаты выполнения программы для генератора rc4

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 98, 1: 109, 2: 96, 3: 109, 4: 95, 5: 113, 6: 122, 7: 93, 8: 93, 9: 113, 10: 114, 11: 110, 12: 102, 13: 85, 14: 98, 15: 89, 16: 98, 17: 109, 18: 113, 19: 106, 20: 125, 21: 95, 22: 100, 23: 103}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 23.295.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.0038.

```

Рисунок 62 – Результаты выполнения программы для генератора rsa

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:e
Уровень значимости: 0.05.
Эмпирическое распределение по частотам: {0: 98, 1: 117, 2: 112, 3: 97, 4: 85, 5: 100, 6: 101, 7: 99, 8: 105, 9: 103, 10: 92, 11: 117, 12: 113, 13: 113, 14: 106, 15: 96, 16: 98, 17: 96, 18: 106, 19: 94, 20: 99, 21: 114, 22: 122, 23: 100}.
Теоретическое значение вероятности для каждой категории: 0.042. Теоретическое количество попаданий в каждую категорию: 105.0.
Критическое значение для критерия хи-квадрат с 24 степенями свободы: 36.415.
Найденное значение хи-квадрат: 18.981.
Представленная последовательность удовлетворяет критерию перестановок: True.

Мат. ожидание последовательности: 0.5049. Её среднеквадратичное отклонение: 0.2891.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.0014.

```

Рисунок 63 – Результаты выполнения программы для генератора bbs

### 1.3.6 Критерий монотонности

Описание критерия:

Последовательность можно проверить на предмет равномерности распределения монотонных серий чисел.

Суть метода в том, чтобы проверить длины всех восходящих серий  $c_i$  в последовательности и подсчитать для них статистику.

Для решения проблемы чередования длинных серий с короткими сериями можно сделать следующее:

1. «Выбрасываем» элемент последовательности, который следует непосредственно за серией.
2. Если  $x_j$  больше  $x_{j+1}$ , то начнем следующую серию с  $x_{j+2}$ .
3. Мы получаем серии, длины которых независимы и, поэтому, можно использовать критерий хи-квадрат.

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_lc /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {3: 401, 2: 1220, 1: 1819, 5: 19, 4: 161, 6: 25}.
Теоретические значения длин серий: [1822.5, 1214.879, 455.625, 121.379, 25.151, 4.374].
Критическое значение для критерия хи-квадрат с 6 степенями свободы: 12.592.
Найденное значение хи-квадрат: 118.279.
Представленная последовательность удовлетворяет критерию монотонности: False.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0028.
```

Рисунок 64 – Результаты выполнения программы для генератора lc

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_add /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {1: 1861, 2: 1227, 3: 444, 4: 131, 5: 24, 6: 3}.
Теоретические значения длин серий: [1845.0, 1229.877, 461.25, 122.877, 25.461, 4.428].
Критическое значение для критерия хи-квадрат с 6 степенями свободы: 12.592.
Найденное значение хи-квадрат: 1.872.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2886.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0003.
```

Рисунок 65 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_5p /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {2: 1243, 1: 1832, 3: 456, 5: 32, 4: 112, 7: 2, 6: 2}.
Теоретические значения длин серий: [1839.5, 1226.211, 459.875, 122.511, 25.385, 4.415, 0.736].
Критическое значение для критерия хи-квадрат с 7 степенями свободы: 14.067.
Найденное значение хи-квадрат: 6.41.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.495. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0101; среднеквадратичного отклонения: 0.0028.
```

Рисунок 66 – Результаты выполнения программы для генератора 5р

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_lfsr /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {2: 1152, 1: 1893, 3: 471, 4: 136, 5: 23, 6: 8}.
Теоретические значения длин серий: [1841.5, 1227.544, 460.375, 122.644, 25.413, 4.42].
Критическое значение для критерия хи-квадрат с 6 степенями свободы: 12.592.
Найденное значение хи-квадрат: 10.918.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.4949. Её среднеквадратичное отклонение: 0.2902.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0103; среднеквадратичного отклонения: 0.0052.

```

Рисунок 67 – Результаты выполнения программы для генератора lfsr

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_nfsr /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {1: 1754, 3: 466, 2: 1262, 5: 36, 4: 118, 6: 5}.
Теоретические значения длин серий: [1820.5, 1213.545, 455.125, 121.245, 25.123, 4.369].
Критическое значение для критерия хи-квадрат с 6 степенями свободы: 12.592.
Найденное значение хи-квадрат: 9.511.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.4991. Её среднеквадратичное отклонение: 0.2884.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0018; среднеквадратичного отклонения: 0.001.

```

Рисунок 68 – Результаты выполнения программы для генератора nfsr

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {2: 1194, 3: 473, 1: 1833, 4: 124, 7: 2, 5: 31, 6: 5}.
Теоретические значения длин серий: [1831.0, 1220.545, 457.75, 121.945, 25.268, 4.394, 0.732].
Критическое значение для критерия хи-квадрат с 7 степенями свободы: 14.067.
Найденное значение хи-квадрат: 4.703.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.5012. Её среднеквадратичное отклонение: 0.2877.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0024; среднеквадратичного отклонения: 0.0035.

```

Рисунок 69 – Результаты выполнения программы для генератора mt

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {1: 1850, 2: 1197, 3: 468, 4: 136, 5: 19, 6: 5, 7: 1}.
Теоретические значения длин серий: [1838.0, 1225.211, 459.5, 122.411, 25.364, 4.411, 0.735].
Критическое значение для критерия хи-квадрат с 7 степенями свободы: 14.067.
Найденное значение хи-квадрат: 4.165.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.503. Её среднеквадратичное отклонение: 0.2903.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.006; среднеквадратичного отклонения: 0.0055.

```

Рисунок 70 – Результаты выполнения программы для генератора rc4

```

○ [alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {1: 1801, 2: 1227, 3: 479, 4: 114, 5: 32, 7: 1, 6: 4}.
Теоретические значения длин серий: [1829.0, 1219.211, 457.25, 121.811, 25.24, 4.39, 0.732].
Критическое значение для критерия хи-квадрат с 7 степенями свободы: 14.067.
Найденное значение хи-квадрат: 3.957.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.4992. Её среднеквадратичное отклонение: 0.2876.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0016; среднеквадратичного отклонения: 0.0038.

```

Рисунок 71 – Результаты выполнения программы для генератора rsa

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_bbs /c:f
Уровень значимости: 0.05.
Эмпирические значения длин серий: {1: 1855, 3: 452, 2: 1222, 4: 120, 5: 29, 6: 6}.
Теоретические значения длин серий: [1842.0, 1227.877, 460.5, 122.677, 25.42, 4.421].
Критическое значение для критерия хи-квадрат с 6 степенями свободы: 12.592.
Найденное значение хи-квадрат: 1.403.
Представленная последовательность удовлетворяет критерию монотонности: True.

Мат. ожидание последовательности: 0.5049. Её среднеквадратичное отклонение: 0.2891.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0097; среднеквадратичного отклонения: 0.0014.
```

Рисунок 72 – Результаты выполнения программы для генератора bbs

### 1.3.7 Критерий конфликтов

Описание критерия:

Предположим, нам нужно оценить последовательность случайных чисел, в которой число величин в последовательности намного меньше числа категорий. В этом случае критерий хи-квадрат не применим, но можно использовать критерий конфликтов.

Предположим, что у нас  $m$  урн и  $n$  шаров, причем  $m$  значительно больше  $n$ . Если разместить шары в урнах наугад, то некоторые урны останутся пустыми, а в некоторых будет более одного шара. Когда в одну урну попадает больше одного шара, то говорят, что произошел «конфликт». Критерий конфликтов состоит в подсчете и оценке количества конфликтов.

Рассмотрим пример, когда  $m = 2^{20}$ , а  $n = 2^{14}$ . В среднем, число урн, приходящихся на один шар – 64. Вероятность того, что в конкретную урну попадет ровно  $k$  шаров, равна

$$p_k = \binom{n}{k} m^{-k} (1 - m^{-1})^{n-k},$$

отсюда, среднее число конфликтов  $K$  в урне вычисляется по формуле

$$\begin{aligned} K &= \sum_{k \leq 1} (k-1)p_k = \sum_{k \leq 1} kp_k - \sum_{k \leq 1} p_k = \frac{n}{m} - 1 + p_0 \\ p_0 &= (1 - m^{-1})^n = 1 - nm^1 + \binom{n}{2} m^{-2} \\ K &\rightarrow \frac{n^2}{2m} = 128 \end{aligned}$$

Приведем результаты выполнения программы для каждого из генераторов:

```
[alse0722@gavno lab3]$ python3 int.py /f:st_lc /c:g
Было найдено 108 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 42. Само значение m: 42000.
Параметр нормирования d: 4. Количество возникших конфликтов: 8.
Таблица процентных точек: [(4, 0.008), (6, 0.049), (8, 0.164), (11, 0.483), (13, 0.705), (17, 0.947), (19, 0.983)].

Размерность вектора V_j: 18. Количество векторов: 555. Множитель для m: 23. Само значение m: 12765.
Параметр нормирования d: 8. Количество возникших конфликтов: 11.
Таблица процентных точек: [(4, 0.007), (6, 0.045), (9, 0.246), (11, 0.473), (13, 0.699), (17, 0.947), (19, 0.984)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 72. Само значение m: 72000.
Параметр нормирования d: 6. Количество возникших конфликтов: 4.
Таблица процентных точек: [(1, 0.008), (2, 0.031), (4, 0.179), (6, 0.463), (8, 0.743), (10, 0.91), (13, 0.989)].

Размерность вектора V_j: 20. Количество векторов: 500. Множитель для m: 34. Само значение m: 17000.
Параметр нормирования d: 3. Количество возникших конфликтов: 4.
Таблица процентных точек: [(1, 0.005), (2, 0.022), (4, 0.145), (6, 0.407), (8, 0.696), (11, 0.937), (13, 0.985)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 29. Само значение m: 19314.
Параметр нормирования d: 3. Количество возникших конфликтов: 12.
Таблица процентных точек: [(3, 0.003), (5, 0.028), (8, 0.197), (10, 0.417), (12, 0.653), (16, 0.935), (19, 0.989)].

Мат. ожидание последовательности: 0.4922. Её среднеквадратичное отклонение: 0.2894.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0158; среднеквадратичного отклонения: 0.0024.
```

Рисунок 73 – Результаты выполнения программы для генератора lc

```
[alse0722@gavno lab3]$ python3 int.py /f:st_add /c:g
Было найдено 121 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 74. Само значение m: 82214.
Параметр нормирования d: 4. Количество возникших конфликтов: 5.
Таблица процентных точек: [(1, 0.005), (2, 0.02), (5, 0.243), (6, 0.381), (8, 0.667), (11, 0.925), (13, 0.98)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 54. Само значение m: 54000.
Параметр нормирования d: 3. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.005), (4, 0.047), (6, 0.186), (8, 0.429), (10, 0.684), (13, 0.919), (16, 0.988)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 70. Само значение m: 87500.
Параметр нормирования d: 4. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.007), (3, 0.022), (6, 0.215), (8, 0.471), (10, 0.721), (13, 0.934), (15, 0.981)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 50. Само значение m: 62500.
Параметр нормирования d: 4. Количество возникших конфликтов: 10.
Таблица процентных точек: [(4, 0.005), (6, 0.035), (9, 0.205), (11, 0.414), (14, 0.736), (17, 0.923), (20, 0.985)].

Размерность вектора V_j: 16. Количество векторов: 625. Множитель для m: 39. Само значение m: 24375.
Параметр нормирования d: 2. Количество возникших конфликтов: 6.
Таблица процентных точек: [(1, 0.003), (3, 0.042), (5, 0.193), (7, 0.46), (9, 0.727), (12, 0.943), (14, 0.985)].

Мат. ожидание последовательности: 0.4921. Её среднеквадратичное отклонение: 0.2887.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0161; среднеквадратичного отклонения: 0.0.
```

Рисунок 74 – Результаты выполнения программы для генератора add

```
[alse0722@gavno lab3]$ python3 int.py /f:st_5p /c:g
Было найдено 94 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 4 из них будут выведены на экран.
Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 40. Само значение m: 40000.
Параметр нормирования d: 3. Количество возникших конфликтов: 9.
Таблица процентных точек: [(4, 0.005), (6, 0.035), (9, 0.206), (11, 0.416), (14, 0.739), (17, 0.925), (20, 0.986)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 63. Само значение m: 78750.
Параметр нормирования d: 4. Количество возникших конфликтов: 7.
Таблица процентных точек: [(2, 0.003), (4, 0.031), (7, 0.23), (9, 0.474), (11, 0.714), (14, 0.926), (17, 0.988)].

Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 28. Само значение m: 19992.
Параметр нормирования d: 2. Количество возникших конфликтов: 13.
Таблица процентных точек: [(4, 0.004), (6, 0.03), (9, 0.189), (11, 0.393), (14, 0.72), (18, 0.95), (20, 0.984)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 43. Само значение m: 43000.
Параметр нормирования d: 3. Количество возникших конфликтов: 9.
Таблица процентных точек: [(4, 0.01), (5, 0.026), (8, 0.184), (10, 0.396), (13, 0.733), (16, 0.926), (19, 0.987)].

Мат. ожидание последовательности: 0.4951. Её среднеквадратичное отклонение: 0.2895.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0028.
```

Рисунок 75 – Результаты выполнения программы для генератора 5p

```
[alse0722@gavno lab3]$ python3 int.py /f:st_lfsr /c:g
Было найдено 85 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 39. Само значение m: 25974.
Параметр нормирования d: 2. Количество возникших конфликтов: 9.
Таблица процентных точек: [(2, 0.009), (3, 0.029), (5, 0.149), (7, 0.388), (9, 0.66), (12, 0.915), (15, 0.988)].

Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 40. Само значение m: 28560.
Параметр нормирования d: 2. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.006), (3, 0.022), (6, 0.217), (8, 0.475), (10, 0.727), (13, 0.937), (15, 0.983)].

Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 79. Само значение m: 87769.
Параметр нормирования d: 4. Количество возникших конфликтов: 4.
Таблица процентных точек: [(1, 0.007), (2, 0.029), (4, 0.171), (6, 0.449), (8, 0.731), (11, 0.948), (13, 0.988)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 26. Само значение m: 26000.
Параметр нормирования d: 3. Количество возникших конфликтов: 15.
Таблица процентных точек: [(9, 0.008), (11, 0.032), (15, 0.21), (18, 0.47), (21, 0.732), (25, 0.933), (28, 0.983)].

Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 37. Само значение m: 26418.
Параметр нормирования d: 2. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.004), (4, 0.037), (6, 0.156), (8, 0.382), (11, 0.75), (14, 0.941), (16, 0.983)].

Мат. ожидание последовательности: 0.4949. Её среднеквадратичное отклонение: 0.2903.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0103; среднеквадратичного отклонения: 0.0055.
```

Рисунок 76 – Результаты выполнения программы для генератора lfsr

```
[alse0722@gavno lab3]$ python3 int.py /f:st_nfsr /c:g
Было найдено 102 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 17. Само значение m: 18887.
Параметр нормирования d: 3. Количество возникших конфликтов: 32.
Таблица процентных точек: [(19, 0.007), (22, 0.035), (27, 0.205), (31, 0.472), (35, 0.744), (40, 0.937), (44, 0.986)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 62. Само значение m: 77500.
Параметр нормирования d: 4. Количество возникших конфликтов: 7.
Таблица процентных точек: [(3, 0.01), (4, 0.028), (7, 0.216), (9, 0.454), (11, 0.696), (14, 0.918), (17, 0.986)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 76. Само значение m: 95000.
Параметр нормирования d: 4. Количество возникших конфликтов: 7.
Таблица процентных точек: [(1, 0.002), (3, 0.036), (5, 0.173), (7, 0.427), (9, 0.695), (12, 0.929), (14, 0.98)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 92. Само значение m: 115000.
Параметр нормирования d: 4. Количество возникших конфликтов: 7.
Таблица процентных точек: [(1, 0.009), (2, 0.034), (4, 0.194), (6, 0.485), (7, 0.634), (10, 0.919), (12, 0.98)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 48. Само значение m: 48000.
Параметр нормирования d: 3. Количество возникших конфликтов: 11.
Таблица процентных точек: [(3, 0.007), (4, 0.022), (7, 0.188), (9, 0.415), (11, 0.659), (15, 0.941), (17, 0.982)].

Мат. ожидание последовательности: 0.4988. Её среднеквадратичное отклонение: 0.2885.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0024; среднеквадратичного отклонения: 0.0007.
```

Рисунок 77 – Результаты выполнения программы для генератора nfsr

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_mt /c:g
было найдено 80 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 24. Само значение m: 17136.
Параметр нормирования d: 2. Количество возникших конфликтов: 16.
Таблица процентных точек: [(6, 0.008), (8, 0.041), (11, 0.202), (14, 0.5), (16, 0.701), (20, 0.936), (23, 0.987)].

Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 20. Само значение m: 22220.
Параметр нормирования d: 3. Количество возникших конфликтов: 28.
Таблица процентных точек: [(15, 0.006), (18, 0.035), (23, 0.23), (26, 0.449), (30, 0.743), (35, 0.943), (39, 0.989)].

Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 72. Само значение m: 79992.
Параметр нормирования d: 4. Количество возникших конфликтов: 5.
Таблица процентных точек: [(1, 0.004), (2, 0.017), (5, 0.221), (7, 0.499), (8, 0.639), (11, 0.912), (14, 0.988)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 67. Само значение m: 83750.
Параметр нормирования d: 4. Количество возникших конфликтов: 9.
Таблица процентных точек: [(2, 0.005), (4, 0.045), (6, 0.18), (8, 0.418), (10, 0.674), (13, 0.914), (16, 0.986)].

Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 56. Само значение m: 70000.
Параметр нормирования d: 4. Количество возникших конфликтов: 9.
Таблица процентных точек: [(3, 0.004), (5, 0.034), (8, 0.221), (10, 0.448), (12, 0.68), (16, 0.943), (18, 0.982)].

Мат. ожидание последовательности: 0.5012. Её среднеквадратичное отклонение: 0.2879.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0024; среднеквадратичного отклонения: 0.0028.
```

Рисунок 78 – Результаты выполнения программы для генератора mt

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_rc4 /c:g
было найдено 76 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 50. Само значение m: 50000.
Параметр нормирования d: 3. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.003), (4, 0.029), (7, 0.224), (9, 0.466), (11, 0.707), (14, 0.923), (17, 0.988)].

Размерность вектора V_j: 16. Количество векторов: 625. Множитель для m: 45. Само значение m: 28125.
Параметр нормирования d: 2. Количество возникших конфликтов: 5.
Таблица процентных точек: [(1, 0.007), (2, 0.031), (4, 0.18), (6, 0.465), (8, 0.746), (10, 0.913), (13, 0.99)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 42. Само значение m: 27972.
Параметр нормирования d: 2. Количество возникших конфликтов: 5.
Таблица процентных точек: [(1, 0.003), (3, 0.044), (5, 0.201), (7, 0.472), (9, 0.737), (12, 0.946), (14, 0.987)].

Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 18. Само значение m: 12852.
Параметр нормирования d: 2. Количество возникших конфликтов: 21.
Таблица процентных точек: [(9, 0.005), (12, 0.044), (16, 0.25), (18, 0.425), (21, 0.695), (26, 0.946), (29, 0.987)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 51. Само значение m: 51000.
Параметр нормирования d: 3. Количество возникших конфликтов: 10.
Таблица процентных точек: [(2, 0.003), (4, 0.033), (7, 0.242), (9, 0.491), (11, 0.729), (14, 0.932), (17, 0.99)].

Мат. ожидание последовательности: 0.5027. Её среднеквадратичное отклонение: 0.2904.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.0054; среднеквадратичного отклонения: 0.0059.
```

Рисунок 79 – Результаты выполнения программы для генератора rc4

```
[alse0722@gavno lab3]$ python3 rnt.py /f:st_good_rsa /c:g
было найдено 83 наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 61. Само значение m: 61000.
Параметр нормирования d: 3. Количество возникших конфликтов: 9.
Таблица процентных точек: [(1, 0.002), (3, 0.037), (5, 0.175), (7, 0.431), (9, 0.7), (12, 0.931), (14, 0.981)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 27. Само значение m: 17982.
Параметр нормирования d: 2. Количество возникших конфликтов: 10.
Таблица процентных точек: [(4, 0.006), (6, 0.038), (9, 0.222), (11, 0.44), (13, 0.666), (17, 0.935), (20, 0.989)].

Размерность вектора V_j: 10. Количество векторов: 1000. Множитель для m: 64. Само значение m: 64000.
Параметр нормирования d: 3. Количество возникших конфликтов: 9.
Таблица процентных точек: [(1, 0.003), (3, 0.048), (5, 0.211), (7, 0.485), (9, 0.747), (12, 0.949), (14, 0.987)].

Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 20. Само значение m: 22220.
Параметр нормирования d: 3. Количество возникших конфликтов: 28.
Таблица процентных точек: [(15, 0.006), (18, 0.035), (23, 0.23), (26, 0.449), (30, 0.743), (35, 0.943), (39, 0.989)].

Размерность вектора V_j: 14. Количество векторов: 714. Множитель для m: 32. Само значение m: 22848.
Параметр нормирования d: 2. Количество возникших конфликтов: 10.
Таблица процентных точек: [(3, 0.004), (5, 0.034), (8, 0.224), (10, 0.454), (12, 0.688), (16, 0.947), (18, 0.984)].

Мат. ожидание последовательности: 0.499. Её среднеквадратичное отклонение: 0.2875.
Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.
Относительные погрешности мат. ожидания: 0.002; среднеквадратичного отклонения: 0.0042.
```

Рисунок 80 – Результаты выполнения программы для генератора rsa

```
[false0722@gavno lab3]$ python3 znt.py /f:st_good_bbs /c:g
Было найдено 75 наборов параметров, при которых представленная последовательность удовлетворяет критерий конфликтов.
Случайные 5 из них будут выведены на экран.
Размерность вектора V_j: 8. Количество векторов: 1250. Множитель для m: 40. Само значение m: 50000.
Параметр нормирования d: 4. Количество возникших конфликтов: 14.
Таблица процентных точек: [(6, 0.005), (8, 0.027), (12, 0.225), (14, 0.415), (17, 0.709), (21, 0.934), (24, 0.986)].

Размерность вектора V_j: 9. Количество векторов: 1111. Множитель для m: 79. Само значение m: 87769.
Параметр нормирования d: 4. Количество возникших конфликтов: 4.
Таблица процентных точек: [(1, 0.007), (2, 0.029), (4, 0.171), (6, 0.449), (8, 0.731), (11, 0.948), (13, 0.988)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 35. Само значение m: 23310.
Параметр нормирования d: 2. Количество возникших конфликтов: 6.
Таблица процентных точек: [(2, 0.004), (4, 0.04), (6, 0.167), (8, 0.4), (10, 0.658), (14, 0.947), (16, 0.985)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 48. Само значение m: 31968.
Параметр нормирования d: 2. Количество возникших конфликтов: 6.
Таблица процентных точек: [(1, 0.008), (2, 0.031), (4, 0.18), (6, 0.466), (8, 0.747), (10, 0.913), (13, 0.99)].

Размерность вектора V_j: 15. Количество векторов: 666. Множитель для m: 45. Само значение m: 29970.
Параметр нормирования d: 2. Количество возникших конфликтов: 6.
Таблица процентных точек: [(1, 0.005), (2, 0.022), (4, 0.141), (6, 0.398), (8, 0.686), (11, 0.933), (13, 0.983)].
```

Мат. ожидание последовательности: 0.505. Её среднеквадратичное отклонение: 0.2892.  
 Эталонные значения мат. ожидания: 0.5; среднеквадратичного отклонения: 0.2887.  
 Относительные погрешности мат. ожидания: 0.0099; среднеквадратичного отклонения: 0.0017.

Рисунок 81 – Результаты выполнения программы для генератора bbs

### 1.3.8 Результаты проверки всех критериев для последовательностей, сгенерированных каждым генератором

В следующей таблице представлены результаты проверки для всех сгенерированных последовательностей по каждому из рассмотренных выше критериев. Зеленый цвет - критерий пройден, красный - не пройден.

	$\chi^2$	серий	интервал.	разбиений	перест.	монот.	конфл.
lc	3.2297	3.353	13.313	2.037	119.287	46.48	148
add	24.7563	15.019	9.321	9.402	18.99	2.279	106
5p	9.5378	65.929	12.985	3.94	131.638	3991940	378
lfsr	12.902	2097.11	30.225	52.143	481.057	371.465	211
nfsr	7.7395	3947.99	358.885	677.402	1584.28	43547900	333
mt	13.916	12.808	5.571	2.233	20.038	6.745	115
rc4	822.272	135.25	30.929	4.392	26.81	10.65	130
rsa	164.639	57.526	29.857	3.496	11.238	18.575	87
bbs	17.7563	16.006	6.135	4.856	18.029	1.422	99

## ПРИЛОЖЕНИЕ А

### Исходный код программы проверки критериев

```
import scipy
import sys
import math
import random
import numpy as np
import scipy.stats
import scipy.special
from sympy.functions.combinatorial.numbers import stirling
import matplotlib.pyplot as plt

sqrt_two = math.sqrt(2)
crit_unique_values = 49
eps = 0.001

def compute_sd(seq, l_seq, expected=False):
    if type(expected) == type(True):
        expected = compute_expected(seq, l_seq)
    return math.sqrt(sum([(seq[j] - expected)** 2 for j in range(l_seq)]) / l_seq)

def compute_expected(seq, l_seq):
    return sum(seq) / l_seq

def uniform(seq, l_seq, d=64, targeted=True):
    seq = list(map(lambda elem : math.floor(elem * d), seq))
    alpha = .05
    critical_value = round(scipy.stats.chi2.ppf(1 - alpha, d - 1), 3)
    est, chi = l_seq / d, 0

    if targeted:
        print(f"Уровень значимости: {alpha}.")
        print(f"Критическое значение для критерия равномерности с {d - 1} степенями свободы: {critical_value}.")
        for i in range(d):
            chi += ((seq.count(i) - est)** 2) / est
        chi = round(chi, 3)
        print(f"Вычисленное значение хи-квадрат: {chi}.")
        print(f"Представленная последовательность удовлетворяет критерию равномерности: {chi < critical_value}.")

    return seq

def series(seq, l_seq):
    d, matches = 4, {}
    d_sq = d * d
```

```

l_unique_values = len(np.unique(seq))

for i in range(d):
    for j in range(d):
        matches[(i, j)] = 0
seq = uniform(seq, l_seq, d, targeted=False)
alpha, chi = .05, 0
critical_value, est = round(scipy.stats.chi2.ppf(1 - alpha, d_sq), 3), round(l_seq / (2 * d_sq), 3)

for i in range(l_seq // 2):
    matches[(seq[2 * i], seq[2 * i + 1])] += 1

print(f"Совпадения пар последовательных чисел: {matches}.")
print(f"Теоретическое количество попаданий в каждую категорию: {est}.")
print(f"Уровень значимости: {alpha}.")
print(f"Параметр d принимает значение: {d}.")
print(f"Критическое значение для критерия серий с {d_sq} степенями свободы: {critical_value}.")
for key in matches.keys():
    chi += ((matches[key] - est) ** 2) / est
chi = round(chi, 3)
print(f"Вычисленное значение хи-квадрат: {chi}.")
print(f"Представленная последовательность удовлетворяет критерию серий: {chi < critical_value}.")
if l_unique_values < crit_unique_values:
    print(f"Из-за маленького периода ({l_unique_values}) предложенная последовательность не может удовлетворять критерию серий.")

def intervals(seq, l_seq):

    t_lower, t_upper, n_lower, n_upper = 8, 12, 1000, 2000
    _alpha = .05

    alpha, beta = random.uniform(0, 1), random.uniform(0, 1)
    while not (0.2 < abs(alpha - beta) < 0.4):
        alpha, beta = random.uniform(0, 1), random.uniform(0, 1)
    if beta < alpha: alpha, beta = beta, alpha
    alpha, beta = round(alpha, 3), round(beta, 3)
    p = beta - alpha

    s, count, chi, best_crit = 0, {}, math.inf, math.inf
    best_len, best_ints = math.inf, math.inf
    best_probs, best_count = list(), list()

    for t in range(t_lower, t_upper):
        for n in range(n_lower, n_upper):
            s, count = 0, {}
            for k in range(t + 1): count[k] = 0
            r = 0
            for j in range(l_seq):
                if alpha <= seq[j] < beta:
                    if r >= t: count[t] += 1
                    else: count[r] += 1
                    s += 1
                if s < n: r = 0
                else: break
            else: r += 1
            if s == n and r >= t and r < t + 1 and count[t] >= 0 and count[r] >= 0 and abs(count[t] - est) < best_crit:
                best_len, best_ints = t, r
                best_probs, best_count = list(count.values()), list(count.keys())
                best_crit = abs(count[t] - est)

```

```

probs, chi_acc = [round(p * ((1 - p) ** _r), 3) for _r in range(t)], 0
probs.append(round((1 - p) ** t, 4))
critical_value = round(scipy.stats.chi2.ppf(1 - _alpha, t + 1), 3)
for j in range(len(probs)):
    est = n * probs[j]
    if est == 0: break
    chi_acc += ((count[j] - est) ** 2) / est
if chi_acc < chi:
    chi, best_crit = chi_acc, critical_value
    best_len, best_ints = t, n
    best_probs, best_count = probs, count

print(f"Уровень значимости: {_alpha}.")
print(f"Критическое значение для критерия хи-квадрат с {best_len + 1} степенями свободы: {best_crit}.")
print(f"Найденное оптимальное значение хи-квадрата: {round(chi, 3)}")
print(f"Это значение было найдено при параметрах t (макс. длина интервала) = {best_len} и n (количество
интервалов) = {best_ints}.")
print(f"Используемые значения границ alpha = {alpha}, beta = {beta}.")
print(f"Представленная последовательность удовлетворяет критерию интервалов: {chi < best_crit}.")
print(f"Пересчитанные вероятности p_r и p_t: {best_probs}.")
print(f"Подсчитанные значения интервалов длиной 0, 1, ..., t - 1 и >= t: {best_count}.")

```

**def partitions(seq, l\_seq, d=8, k=5):**

```

seq = uniform(seq, l_seq, d, targeted=False)
groups_quan, alpha, chi, num_groups = {}, .05, 0, l_seq // k
critical_value = round(scipy.stats.chi2.ppf(1 - alpha, k - 1), 3)
for i in range(k, 0, -1): groups_quan[i] = 0
counter = 0

for i in range(num_groups):
    current_group = seq[(i * k):(i + 1) * k]
    groups_quan[len(list(np.unique(current_group)))] += 1
probs = [round((math.prod([i for i in range(d, d - r, -1)]) / (d ** k)) * stirling(k, r, kind=2), 4) for r in groups_quan.keys()]
chi = sum([(value - probs[i] * num_groups) ** 2) / (probs[i] * num_groups) for i, value in
enumerate(groups_quan.values())])

print(f"Уровень значимости: {alpha}.")
print(f"Получившийся словарь комбинаций (первый элемент соответствует количеству уникальных символов в
группе): {groups_quan}.")
print(f"Вычисленные теоретические вероятности частоты комбинаций: {probs}.")
print(f"Критическое значение для критерия хи-квадрат с {k - 1} степенями свободы: {critical_value}.")
print(f"Найденное значение хи-квадрат: {chi}.")
print(f"Представленная последовательность удовлетворяет критерию разбиений: {chi < critical_value}.")

```

**def permutations(seq, l\_seq, t=4):**

```

df = math.prod([j for j in range(1, t + 1)])
categ_quan = {i: 0 for i in range(df)}
alpha, num_groups = .05, l_seq // t
critical_value = round(scipy.stats.chi2.ppf(1 - alpha, df), 3)

```

```

def perm_analysis(perm):
    r, f = t, 0
    while r > 1:

```

```

s = perm.index(max(perm)) + 1
f = r * f + s - 1
perm[r - 1], perm[s - 1] = perm[s - 1], perm[r - 1]
perm, r = perm[:-1], r - 1

return f

for i in range(num_groups):
    current_group = seq[(i * t):(i + 1) * t]
    if len(np.unique(current_group)) != t: continue
    else: categ_quan[perm_analysis(current_group)] += 1

est, chi = round(1 / df, 3), 0
est_cat = round(est * num_groups, 3)
print(f"Уровень значимости: {alpha}.")
print(f"Эмпирическое распределение по частотам: {categ_quan}.")
print(f"Теоретическое значение вероятности для каждой категории: {est}. Теоретическое количество попаданий в каждую категорию: {est_cat}.")
print(f"Критическое значение для критерия хи-квадрат с {df} степенями свободы: {critical_value}.")
chi = round(sum([(categ_quan[i] - est_cat) ** 2] / est_cat for i in range(df))), 3
print(f"Найденное значение хи-квадрат: {chi}.")
print(f"Представленная последовательность удовлетворяет критерию перестановок: {chi < critical_value}.")

def monotonous(seq, l_seq):
    categ_quan, est = {}, list()
    alpha, chi, i = .05, 0, 0

    current_spree = 1
    while i < l_seq - 1:
        if seq[i] < seq[i + 1]: current_spree += 1
        else:
            if current_spree not in categ_quan:
                categ_quan[current_spree] = 1
            else: categ_quan[current_spree] += 1
            current_spree, i = 1, i + 1
        i += 1
    sum_series = sum([value for value in categ_quan.values()])
    num_categs = len(categ_quan)
    critical_value = round(scipy.stats.chi2.ppf(1 - alpha, num_categs), 3)

    fact_acc = 1
    for i in range(1, num_categs + 1):
        fact_acc *= i
        est.append(round(1 / fact_acc - 1 / (fact_acc * (i + 1)), 4))
    est = [round(sum_series * est[j], 3) for j in range(num_categs)]
    chi = round(sum([(categ_quan[j + 1] - est[j]) ** 2] / est[j] for j in range(num_categs))), 3

    print(f"Уровень значимости: {alpha}.")
    print(f"Эмпирические значения длин серий: {categ_quan}.")
    print(f"Теоретические значения длин серий: {est}.")
    print(f"Критическое значение для критерия хи-квадрат с {num_categs} степенями свободы: {critical_value}.")
    print(f"Найденное значение хи-квадрат: {chi}.")
    print(f"Представленная последовательность удовлетворяет критерию монотонности: {chi < critical_value}.")

```

```

def conflicts(seq, l_seq):
    s_lower, s_upper, d_lower, d_upper, n_to_print = 8, 20, 2, 8, 5
    m_lower, m_upper = 16, 128
    eps = 1e-20
    T_table = (.01, .05, .25, .50, .75, .95, .99, 1.)

    def percent_points(m, n):
        A, conflicts_et_probs = [0] * (n + 1), list()
        A[1] = j_0 = j_1 = 1

        for i in range(n - 1):
            j_1 += 1
            for j in range(j_1, j_0 - 1, -1):
                j_by_m = j / m
                A[j] = j_by_m * A[j] + (1 + 1 / m - j_by_m) * A[j - 1]
                if A[j] < eps:
                    A[j] = 0
                if j == j_1: j_1 -= 1; continue
                if j == j_0: j_0 += 1

        p, t, j = 0, 0, j_0 - 1
        while t != len(T_table) - 1:
            while p <= T_table[t]:
                j += 1
                p += A[j]
            conflicts_et_probs.append((n - j - 1, round(1 - p, 3)))
            t += 1

    return conflicts_et_probs

suitables = list()
for vec_size in range(s_lower, s_upper + 1):
    n_param = l_seq // vec_size
    for d_param in range(d_lower, d_upper + 1):
        seq_normed = uniform(seq, l_seq, d=d_param, targeted=False)
        words, n_conflicts = list(), 0
        for j_index in range(n_param):
            _slice = seq_normed[(j_index * vec_size):(j_index + 1) * vec_size]
            if _slice not in words: words.append(_slice)
            else: n_conflicts += 1
        for m_param in range(m_lower, m_upper + 1):
            m_value = n_param * m_param
            confs_et_probs = percent_points(m_value, n_param)[::-1]
            if n_conflicts == 0 or confs_et_probs[0][0] == -1 or confs_et_probs[0][0] == 0:
                continue
            if confs_et_probs[2][0] <= n_conflicts <= confs_et_probs[-3][0]:
                suitables.append((confs_et_probs, n_conflicts, vec_size, n_param, d_param, m_param, m_value))

l_suits, rand_idxs = len(suitables), list()
for j in range(n_to_print):
    rand_idx = random.randint(0, l_suits - 1)
    if rand_idx not in rand_idxs: rand_idxs.append(rand_idx)
    else:
        while rand_idx in rand_idxs: rand_idx = random.randint(0, l_suits - 1)

```

```

print(f"Было найдено {l_suits} наборов параметров, при которых представленная последовательность удовлетворяет критерию конфликтов.")
print(f"Случайные {len(rand_idxs)} из них будут выведены на экран.")
for j in range(len(rand_idxs)):
    confs_et_probs, n_conflicts, vec_size, n_param, d_param, m_param, m_value = suitables[rand_idxs[j] % l_suits]
    print(f"Размерность вектора V_{j}: {vec_size}. Количество векторов: {n_param}. Множитель для m: {m_param}. Само значение m: {m_value}.")
    print(f"Параметр нормирования d: {d_param}. Количество возникших конфликтов: {n_conflicts}.")
    print(f"Таблица процентных точек: {confs_et_probs}.", end='\n\n')

def chi_wrapper(seq, l_seq, intervals):
    print(f"Используемые диапазоны интервалов: {[round(interval[0], 4), round(interval[1], 4)) for interval in intervals]}")
    ints_len = len(intervals)

    def printer(est, chi_val, chi_crit):
        print(f"Ожидаемое распределение чисел по интервалам: {est}.")
        print(f"Значение хи-квадрат: {chi_val}.")
        print(f"Результат о принятии гипотезы: {'принята' if 0 < chi_val < chi_crit else 'не принята'}.", end='\n')

    actual = [0 for i in range(ints_len)]
    for i in range(l_seq):
        for j in range(ints_len):
            if intervals[j][0] <= seq[i] < intervals[j][1]:
                actual[j] += 1; break

    est = round(l_seq / ints_len, 3)
    chi_st = round(sum([(actual[j] - est) ** 2 / est for j in range(ints_len)]), 4)
    chi_crit = round(scipy.stats.chi2.ppf(1-0.05, df=(ints_len - 1)), 4)

    print(f"Количество степеней свободы: {ints_len - 1}. Уровень значимости: {0.05}.", end=' ')
    print(f"Критическое значение хи-квадрат: {chi_crit}.")
    print(f"Наблюдаемое распределение чисел по интервалам: {actual}.", end='\n')
    printer(est, chi_st, chi_crit)

def draw_params(seq, l_seq):
    steps = (50, 100, 200, 500)
    exp_vals_stepped, _sd_vals_stepped = list(), list()

    for step in steps:
        exp_acc, exp_vals = 0, list()
        sd_vals = list()
        for j in range(l_seq // step):
            idx_l, idx_r = j * step, (j + 1) * step
            _slice_exp = compute_expected(seq[idx_l:idx_r], idx_r)
            exp_acc = exp_acc * idx_l / idx_r + _slice_exp
            exp_vals.append(exp_acc)
            sd_acc = compute_sd(seq[:idx_r], idx_r, exp_acc)
            sd_vals.append(sd_acc)
        exp_vals_stepped.append(exp_vals)
        _sd_vals_stepped.append(sd_vals)

```

```

exp_fig, exp_axs = plt.subplots(nrows=2, ncols=2)
exp_fig.suptitle('Сходимость мат. ожидания для различных значений шага')
_sd_fig, _sd_axs = plt.subplots(nrows=2, ncols=2)
_sd_fig.suptitle('Сходимость среднеквадратичного отклонения для различных значений шага')

for j in range(len(steps)):
    j_bin = bin(j)[2:]
    l_j_bin = len(j_bin)
    stepped = [steps[j] * (k + 1) for k in range(l_seq // steps[j])]
    idx_0 = 0 if l_j_bin == 1 else int(j_bin[0])
    idx_1 = int(j_bin[0]) if l_j_bin == 1 else int(j_bin[1])

    exp_axs[idx_0, idx_1].plot(stepped, exp_vals_stepped[j])
    exp_axs[idx_0, idx_1].set_title(f"Шаг: {steps[j]}")
    _sd_axs[idx_0, idx_1].plot(stepped, _sd_vals_stepped[j])
    _sd_axs[idx_0, idx_1].set_title(f"Шаг: {steps[j]}")

plt.show()

def main():
    file = sys.argv[1][3:]
    criterion = sys.argv[2][3:]

    f = open(file, 'r')
    seq = list(map(float, (f.read()).split(',')))
    l_seq = len(seq)
    unique_values = np.unique(seq)

    if len(unique_values) > crit_unique_values:
        seq = [seq[i] if seq[i] != 1.0 else round(np.random.uniform(), 3) for i in range(l_seq)]
        l_bord, r_bord = min(seq), max(seq)
        num_intervals = math.ceil(1 + 1.4 * math.log(l_seq))
        step = round((r_bord - l_bord) / num_intervals, 4)
        _intervals = [(l_bord + i * step, l_bord + (i + 1) * step) for i in range(num_intervals)]
    else:
        seq = [seq[i] if seq[i] != 1.0 else seq[i] - eps for i in range(l_seq)]
        unique_values = np.unique(seq)
        _intervals = [(value - eps, value + eps) for value in unique_values]

    if criterion=='a':
        chi_wrapper(seq, l_seq, _intervals)
    if criterion=='b':
        series(seq, l_seq)
    if criterion=='c':
        intervals(seq, l_seq)
    if criterion=='d':
        partitions(seq, l_seq)
    if criterion=='e':
        permutations(seq, l_seq)
    if criterion=='f':
        monotonous(seq, l_seq)
    if criterion=='g':
        conflicts(seq, l_seq)

    exp, sd = round(compute_expected(seq, l_seq), 4), round(compute_sd(seq, l_seq), 4)

```

```
exp_th, sd_th = .5, round(math.sqrt(1 / 12), 4)
print()
print(f"Мат. ожидание последовательности: {exp}. Её среднеквадратичное отклонение: {sd}.")
print(f"Эталонные значения мат. ожидания: {exp_th}; среднеквадратичного отклонения: {sd_th}.")
rel_err_exp, rel_err_sd = round(abs(exp - exp_th) / exp, 4), round(abs(sd - sd_th) / sd, 4)
print(f"Относительные погрешности мат. ожидания: {rel_err_exp}; среднеквадратичного отклонения: {rel_err_sd}.")

waiter = input()
draw_params(seq, l_seq)

if __name__ == '__main__':
    main()
```