Міністерство освіти і науки України
Національний технічний університет України «Київський політехнічний інститут імені Ігоря Сікорського"
Факультет інформатики та обчислювальної техніки

Кафедра інформатики та програмної інженерії

Звіт

з лабораторної роботи № 1 з дисципліни
«Моделювання систем»

«ПЕРЕВІРКА ГЕНЕРАТОРА ВИПАДКОВИХ ЧИСЕЛ НА ВІДПОВІСТЬ ЗАКОНУ РОЗПОДІЛУ»

Виконав(ла) *ІП-14 Сергієнко Ю. В.* __________
(шифр, прізвище, ім'я, по батькові)

Перевірив *Дифучин А. Ю.* __________
(прізвище, ім'я, по батькові)

Київ 2024

Комп'ютерний практикум 1

Тема: Перевірка генератора випадкових чисел на відповідність закону розподілу.

Завдання:

- ✓ Згенерувати 10000 випадкових чисел трьома вказаними нижче способами. **45 балів**.
  - Згенерувати випадкове число за формулою $x_i = -\frac{1}{\lambda}\ln\xi_i$, де ξ_i - випадкове число, рівномірно розподілене в інтервалі (0;1). Числа ξ_i можна створювати за допомогою вбудованого в мову програмування генератора випадкових чисел. Перевірити на відповідність експоненційному закону розподілу $F(x) = 1 - e^{-\lambda x}$. Перевірку зробити при різних значеннях λ.
  - Згенерувати випадкове число за формулами:

    $$x_i = \sigma\mu_i + a$$

    $$\mu_i = \sum_{i=1}^{12}\xi_i - 6,$$

    де ξ_i - випадкове число, рівномірно розподілене в інтервалі (0;1). Числа ξ_i можна створювати за допомогою убудованого в мову програмування генератора випадкових чисел. Перевірити на відповідність нормальному закону розподілу:

    $$f(x) = \frac{1}{\sigma\sqrt{2\pi}}\exp\left(-\frac{(x-a)^2}{2\sigma^2}\right).$$

    Перевірку зробити при різних значеннях a і σ.
  - Згенерувати випадкове число за формулою $z_{i+1} = az_i(\mathrm{mod}\, c)$, $x_{+1i} = z_{i+1}/c$, де $a=5^{13}$, $c=2^{31}$. Перевірити на відповідність рівномірному закону розподілу в інтервалі (0;1). Перевірку зробити при різних значеннях параметрів a і c.

- ✓ Для кожного побудованого генератора випадкових чисел побудувати гістограму частот, знайти середнє і дисперсію цих випадкових чисел. По виду гістограми частот визначити вид закону розподілу. **20 балів.**
- ✓ Відповідність заданому закону розподілу перевірити за допомогою критерію згоди χ^2. **30 балів**
- ✓ Зробити висновки щодо запропонованих способів генерування випадкових величин. **5 балів**

Виконання:

Для виконання роботи необхідно згенерувати 10000 випадкових чисел трьома способами. Розглянемо кожний з них:

1. Згенеруємо ряд випадкових чисел за першою формулою за λ = 0.01:

```
def generate_exponential(lambda_val, n):
    generated_list = [random.Random().random() for _ in range(n)]
    return -np.log(generated_list) / lambda_val
```

Для перевірки випадкової величини необхідно визначити довжину інтервалу, знайшовши мінімальне та максимальне значення. За k беремо оптимальне значення при великій к-сті спостережувальних значень — 20. Також обрахуємо середнє значення випадкових чисел та їх дисперсію:

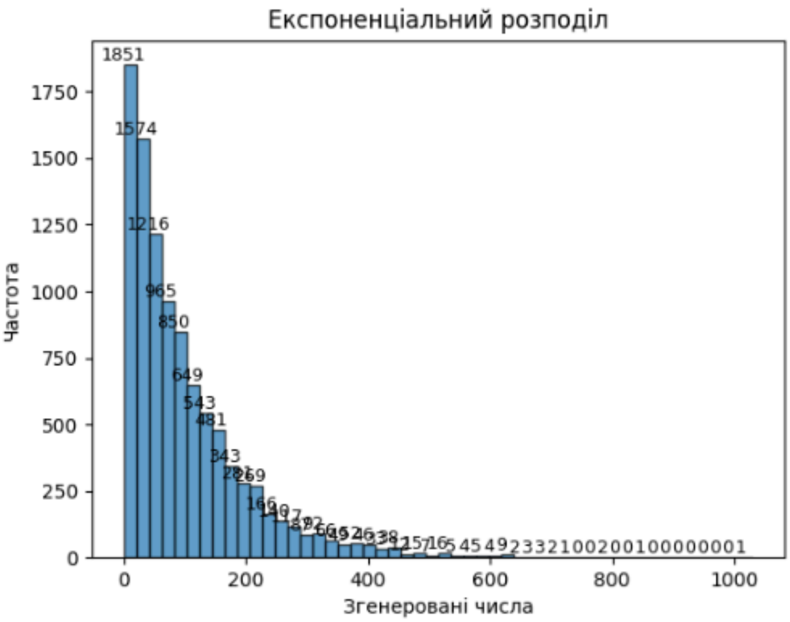
```
min_val = min(sample)
max_val = max(sample)
h = (max_val - min_val) / k
mean = sum(sample) / len(sample)
variance = get_dispersion(mean, sample)
```

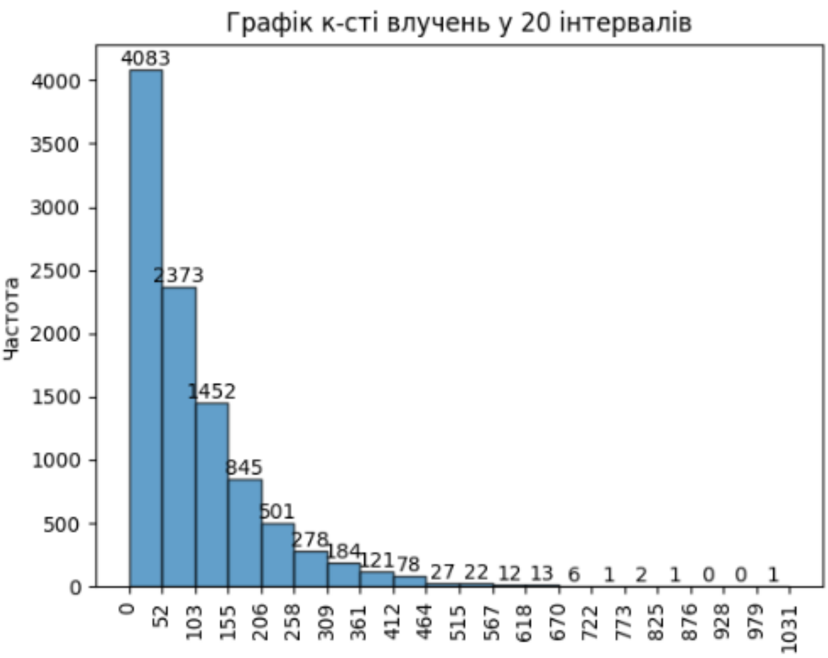
```
def get_dispersion(mean, sample):
    dispersion = 0
    for val in sample:
        dispersion += (val - mean) ** 2
    return dispersion / (len(sample) - 1)
```

Далі, розіб'ємо початковий ряд чисел на k рівних інтервалів та знайдемо к-сті влучень у кожного з них:

```
intervals = list([min_val + i * h for i in range(k + 1)])
frequencies = list(np.histogram(sample, bins=intervals)[0])
```

Виведемо гістограми даного ряду до та після розбиття на інтервали:





З гістограм можна припустити, що це експоненційний розподіл.

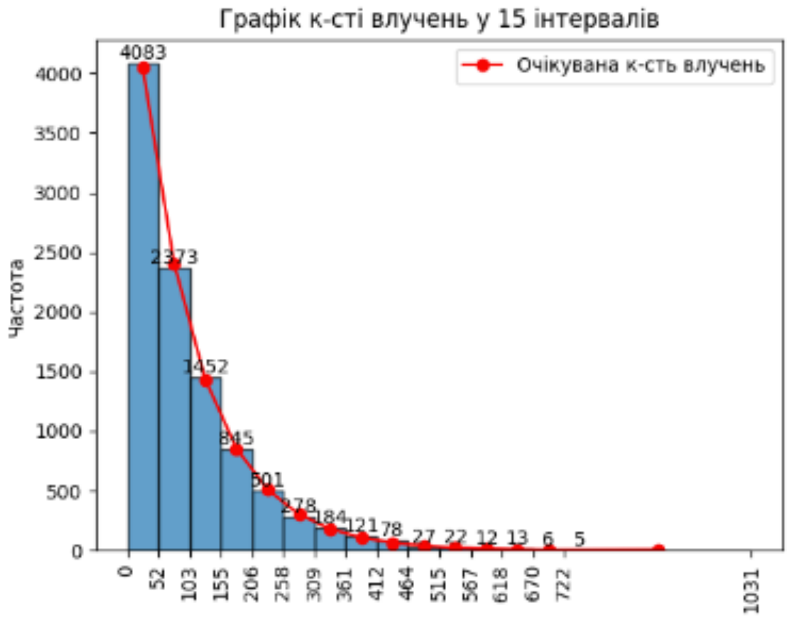
Наступна дія — перевірка к-сті влучань у інтервали. Якщо n < 5, то об'єднуємо його із сусіднім.

```
def merge_intervals(intervals, frequencies):
    i = 0
    while i < len(frequencies):
        if frequencies[i] < 5:
            if i < len(frequencies) - 1:
                frequencies[i] += frequencies[i + 1]
                del frequencies[i + 1]
                del intervals[i + 1]
            elif i > 0:
                # last item
                frequencies[i - 1] += frequencies[i]
                del frequencies[i]
                del intervals[i]
        else:
            i += 1
    return intervals, frequencies
```

Також знайдемо теоретичні к-сті влучень у інтервали для обрахування критерія згоди χ^2. В даному випадку необхідно обчислити вірогідності потрапляння у певний інтервал за допомогою експоненціального закону розподілу.

```
theoretic_frequencies = []
for i in range(len(intervals) - 1):
    theoretic_frequencies.append(
        (expon.cdf(intervals[i + 1], scale=mean) -
         expon.cdf(intervals[i], scale=mean)) * n
    )
```

Розглянемо графік к-сті влучень разом із очікуваними к-стями:



Проведемо перевірку на відповідність закону експоненційного розподілу, обраховуючи χ^2 та порівнюючи його з $\chi^2_{кр}$:

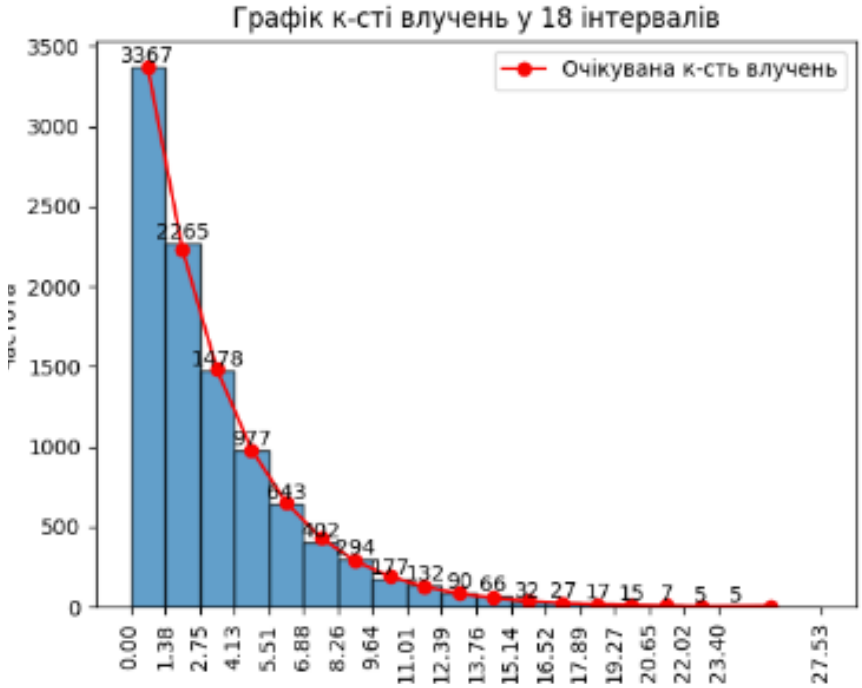
```
def chi_squared_test(frequencies, theoretic_frequencies, count_constants):
    alpha = 0.05  # рівень значущості
    df = len(frequencies) - count_constants - 1  # к-сть інтервалів - к-сть параметрів закону розподілу - 1
    chi2_test = 0
    for i in range(len(frequencies)):
        chi2_test += (frequencies[i] - theoretic_frequencies[i]) ** 2 / theoretic_frequencies[i]
    chi2_kr = chi2.ppf(1 - alpha, df)
```

Розглянемо результати першого досліду:

```
Експоненціальний розподіл:
min: 0.0010829491227072453, max: 1030.760058354396, довжина інтервалу: 51.53794877026367;
середнє μ: 99.39157590890876, дисперсія σ^2: 9883.59068816415
Закон розподілу відповідає спостережуваним значенням випадкової величини, 15.471152112006033 < 22.362032494826938
```

Як бачимо, закон розподілу відповідає спостережуваним значеннями випадкової величини.

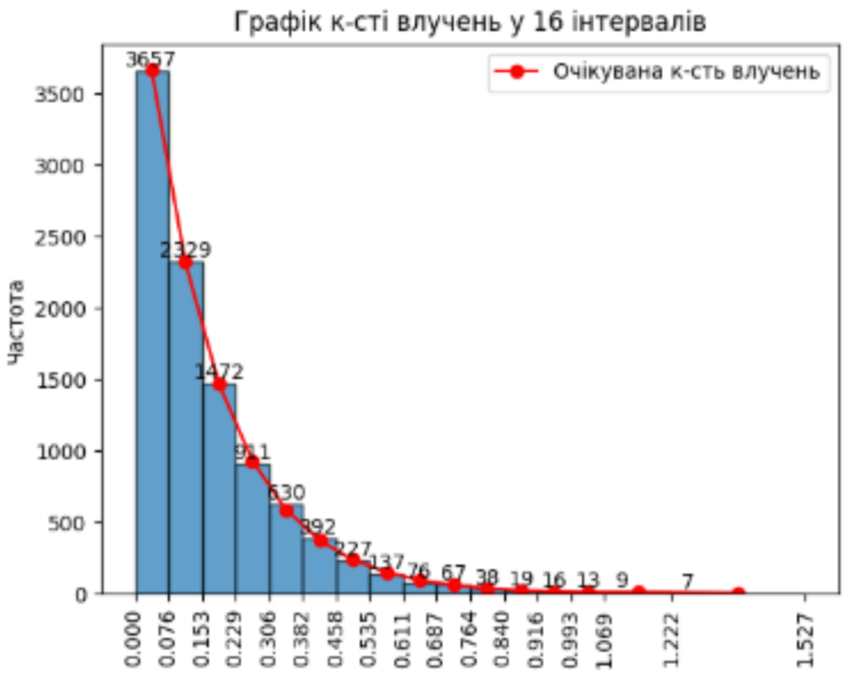
Проведемо експеримент за $\lambda = 0.3$:




```
Експоненціальний розподіл:
min: 5.4430076406675514e-05, max: 27.52969559484545, довжина інтервалу: 1.376482058238452;
середнє μ: 3.3520592666212274, дисперсія σ^2: 11.328566013341016
Закон розподілу відповідає спостережуваним значенням випадкової величини, 9.601532068284202 < 26.296227604864242
```

Як бачимо, розподіл відповідає експоненційному.

Проведемо експеримент за $\lambda = 6$:



```
Експоненціальний розподіл:
min: 2.3730256803776982e-05, max: 1.5274656166233669, довжина інтервалу: 0.07637209431832816;
середнє μ: 0.16727320150518363, дисперсія σ^2: 0.027585992440578362
Закон розподілу відповідає спостережуваним значенням випадкової величини, 12.881260458860602 < 23.68479130484058
```

Бачимо, що розподіл відповідає експоненційному.

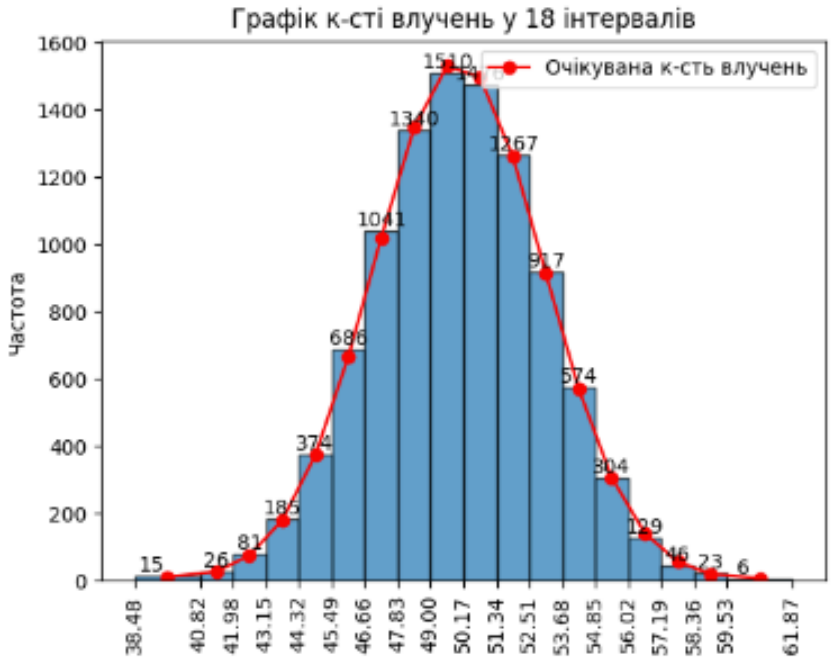
2. Проведемо дослідження другої формули за sigma = 3, a = 50. Згенеруємо початкові дані за формулою:

```
def generate_normal(sigma, a, n):
    return [sigma * (sum(random.Random().random() for _ in range(12)) - 6)
+ a for _ in range(n)]
```

Обрахуємо статистики та інтервали за тим самим принципом. Маємо наступні гістограми по інтервалах:

З гістограм припускаємо, що це нормальний розподіл.

Після об’єднань інтервалів, обрахуємо теоретичні к-сті влучень у інтервали за нормальним законом розподілу та виведемо графік:

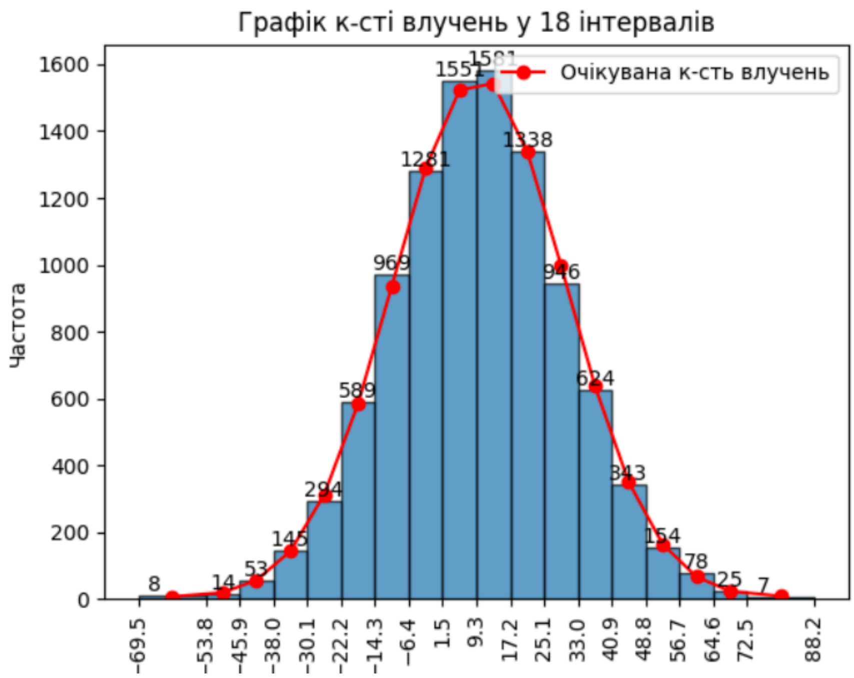


Розглянемо результати другого досліду:

```
Нормальний розподіл:
min: 38.475898798531055, max: 61.868097662389246, довжина інтервалу: 1.1696099431929095;
середнє μ: 49.96142146603087, дисперсія σ^2: 9.06740229807057
Закон розподілу відповідає спостережуваним значенням випадкової величини, 8.351805511488841 < 24.99579013972862
```

Як бачимо, закон розподілу відповідає спостережуваним значеннями випадкової величини.

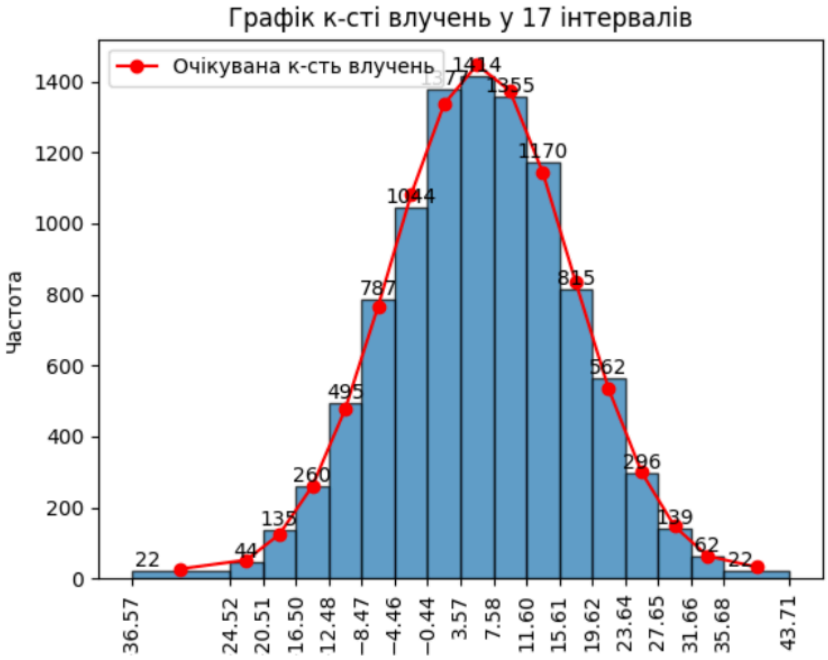
Перевіримо дану формулу за sigma = 20, a = 10. Виведемо гістограму та результат:




```
Нормальний розподіл:
min: -69.54080889062627, max: 88.22770309657706, довжина інтервалу: 7.888425599360167;
середнє μ: 9.870372663568665, дисперсія σ^2: 393.18594108817996
Закон розподілу відповідає спостережуваним значенням випадкової величини, 11.75540432190899 < 24.99579013972862
```

Бачимо, що розподіл відповідає нормальному.

Протестуємо формулу на sigma = 11, a = 6:



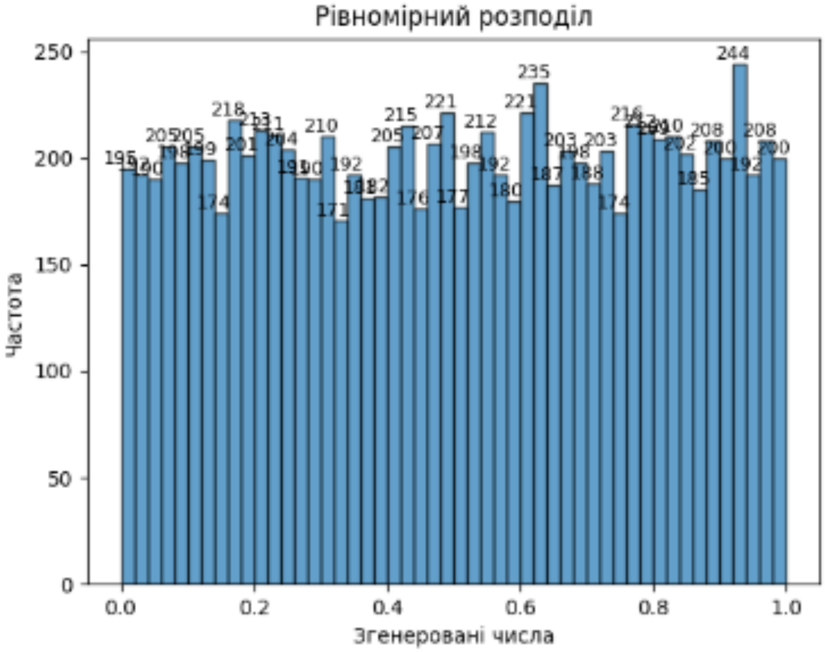

```
Нормальний розподіл:
min: -36.56558078463941, max: 43.70520644809235, довжина інтервалу: 4.013539361636587;
середнє μ: 5.917754197736455, дисперсія σ^2: 119.52080652091193
Закон розподілу відповідає спостережуваним значенням випадкової величини, 13.953422191367025 < 23.68479130484058
```

Гіпотеза справджується.

3. Проведемо дослідження третьої формули. Згенеруємо дані:

```
def generate_uniform(a, c, n):
    generated_list = []
    prev = 1
    for i in range(n):
        prev = (a * prev) % c
        generated_list.append(prev / c)
    return generated_list
```

Після обрахувань статистичних даних та інтервалів, виведемо гістограми:



Можемо припустити, що це рівномірний розподіл.

Обрахуємо теоретичні к-сті влучень в інтервали та виведемо кінцевий графік:

Розглянемо результати третього досліду:

```
Рівномірний розподіл:
min: 4.9410853534936905e-05, max: 0.9999646176584065, довжина інтервалу: 0.04999576034024358;
середнє μ: 0.5041378248833119, дисперсія σ^2: 0.08385396497686254
Закон розподілу відповідає спостережуваним значенням випадкової величини, 21.699999999999996 < 27.587111638275328
```

Як бачимо, закон розподілу відповідає спостережуваним значеннями випадкової величини.

Дослідимо формулу за a = 75 та c = 113:

```
Рівномірний розподіл:
min: 0.008849557522123894, max: 0.9911504424778761, довжина інтервалу: 0.049115044247787606;
середнє μ: 0.5002238938053097, дисперсія σ^2: 0.08182747634227716
Закон розподілу не збігається зі спостережуваними значеннями випадкової величини, 76.368 > 27.587111638275328
```

Як бачимо, закон розподілу не збігається зі спостережуваними значеннями через мале значення c (фактично генерується 113 різних значень, що обмежує випадковість).

Спробуємо a = 10 та c = 3^{18}:


```
Рівномірний розподіл:
min: 2.5811747917131973e-08, max: 0.9999913608079721, довжина інтервалу: 0.04999956674981121;
середнє μ: 0.4997046407277118, дисперсія σ^2: 0.08382300349004428
Закон розподілу відповідає спостережуваним значенням випадкової величини, 23.288 < 27.587111638275328
```

Гіпотеза справджується.

Висновок

У ході виконання лабораторної роботи були досліджені три методи генерації випадкових чисел: експоненційний, нормальний і рівномірний розподіли. Для кожного з них було проведено перевірку відповідності закону розподілу за допомогою критерія χ^2. Усі етапи обчислень включали генерування 10 000 випадкових чисел, побудову гістограм, обчислення середнього значення, дисперсії, частотних характеристик та порівняння результатів із теоретичними очікуваннями.

Експоненційний розподіл: генерація чисел за формулою з параметром λ показала відповідність до експоненційного закону розподілу при різних значеннях λ. Побудовані гістограми підтвердили відповідність згенерованих чисел очікуваному експоненційному розподілу. Після об'єднання інтервалів із недостатньою кількістю спостережень була отримана точна відповідність критерію χ^2.

Нормальний розподіл: Генерація за формулою з параметрами σ та a також продемонструвала очікуваний нормальний розподіл для кількох варіантів параметрів.

Рівномірний розподіл: Генерація чисел за параметрами a та c показала, що при великому значенні c рівномірний розподіл спостерігається чітко, однак при малих значеннях параметра c (наприклад, 113) спостерігався обмежений набір випадкових значень, що вплинуло на відповідність розподілу. Застосування інших значень для a та c (наприклад, $a = 10$, $c = 3^{18}$) забезпечило коректну генерацію випадкових чисел із рівномірним розподілом.

У результаті проведеної роботи були досліджені особливості кожного методу генерації випадкових чисел та підтверджено, що при правильному виборі параметрів кожен із них відповідає очікуваним законам розподілу.

Код програми доступний за посиланням на GitHub.