

ГУАП

КАФЕДРА № 42

ОТЧЕТ  
ЗАЩИЩЕН С ОЦЕНКОЙ \_\_\_\_\_

ПРЕПОДАВАТЕЛЬ

Профессор				Татарникова Т. М.
должность, уч. степень, звание		подпись, дата		инициалы, фамилия

ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №4

**МОДЕЛИРОВАНИЕ ДИСКРЕТНОЙ СЛУЧАЙНОЙ  
ВЕЛИЧИНЫ**

Вариант 5

по курсу: Моделирование систем

РАБОТУ ВЫПОЛНИЛ

СТУДЕНТ ГР. №	4128			Воробьев В. А.
			подпись, дата	инициалы, фамилия

Санкт-Петербург 2024

## СОДЕРЖАНИЕ

<b>1</b>	<b>Постановка задачи</b>	<b>3</b>
1.1	Порядок выполнения работы	3
1.2	Содержание отчета	3
<b>2</b>	<b>Выполнение работы</b>	<b>4</b>
2.1	Теоретические сведения	4
2.2	Результаты моделирования	4
2.3	Анализ	5
<b>3</b>	<b>Вывод</b>	<b>7</b>
	<b>ПРИЛОЖЕНИЕ</b>	<b>8</b>

## 1 Постановка задачи

**Цель работы:** выполнить программную реализацию генератора дискретной случайной величины.

### 1.1 Порядок выполнения работы

1. Выполнить программную реализацию датчика заданной дискретной СВ и сгенерировать выборку из 500 значений дискретной СВ  $x_i$ .
2. Найти эмпирические оценки  $M$  и  $D$  и сравнить их с теоретическими значениями, найденными по (1) и (2).
3. Построить в одном графическом окне две гистограммы: первая – распределение эмпирических вероятностей значений случайной величины  $x$  и вторая – распределение теоретических вероятностей СВ.
4. Дать сравнительную оценку гистограммам распределения эмпирических и теоретических вероятностей случайной величины  $x$ .

Таблица 1.1 - Вариант задания

Параметры	1	2	3	4	5	6	7
$x_i$	-72.4	-68.5	-68.4	-43.5	-6.5	1.2	31.0
$p_i$	0.005	0.240	0.251	0.003	0.271	0.049	0.181

### 1.2 Содержание отчета

1. Листинг программной реализации датчика заданной дискретной СВ.
2. Первые 30 значений  $x_i$ .
3. Результаты эмпирических и теоретических значений  $M$  и  $D$ .
4. Гистограммы распределения эмпирических и теоретических вероятностей дискретной СВ.
5. Выводы по результатам сравнительной оценки распределений эмпирических и теоретических вероятностей дискретной СВ  $x$ .

## 2 Выполнение работы

### 2.1 Теоретические сведения

$$M(x) = \sum_{j=1}^K p_j x_j \quad (1)$$

$$D(x) = \sum_{j=1}^K p_j x_j^2 - M^2(x) \quad (2)$$

### 2.2 Результаты моделирования

Для моделирования были реализованы скрипты на ЯП Python, которые представлены в Приложении и на GitHub (URI - [https://github.com/vladcto/suai-labs/tree/main/6\\_semester/МодСис/4](https://github.com/vladcto/suai-labs/tree/main/6_semester/МодСис/4)).

Результат расчета характеристик представлен на рисунке 2.1.

```
Первые 30 значений: [-68.5, -68.5, -68.5, -68.4, -68.4, 1.2, -6.5, -68.5, -68.5, 31.0, -68.4, -68.5, -68.4, -68.4, -68.4, -68.4, -68.4, -68.5, 31.0, 31.0, -6.5, 31.0, -68.5, 31.0, -68.4, -6.5, -68.5, -68.4, -68.5, -6.5, -6.5]

=== Теоретические ===
Мода = -30.1926
Дисперсия = 1606.2123252400002

=== Эмперические ===
Мода = -30.071099999999998
Дисперсия = 1600.4942407900003
```

Рисунок 2.1 - Расчет характеристик

Была реализована логика для построения графиков, а также подсчета требуемых характеристик.

Для построения графиков было взято 500 точек. Для удобства анализа было решено отобразить распределение случайно величины в формате полупрозрачных бар-чартов. Результат изображен на рисунке 2.2.

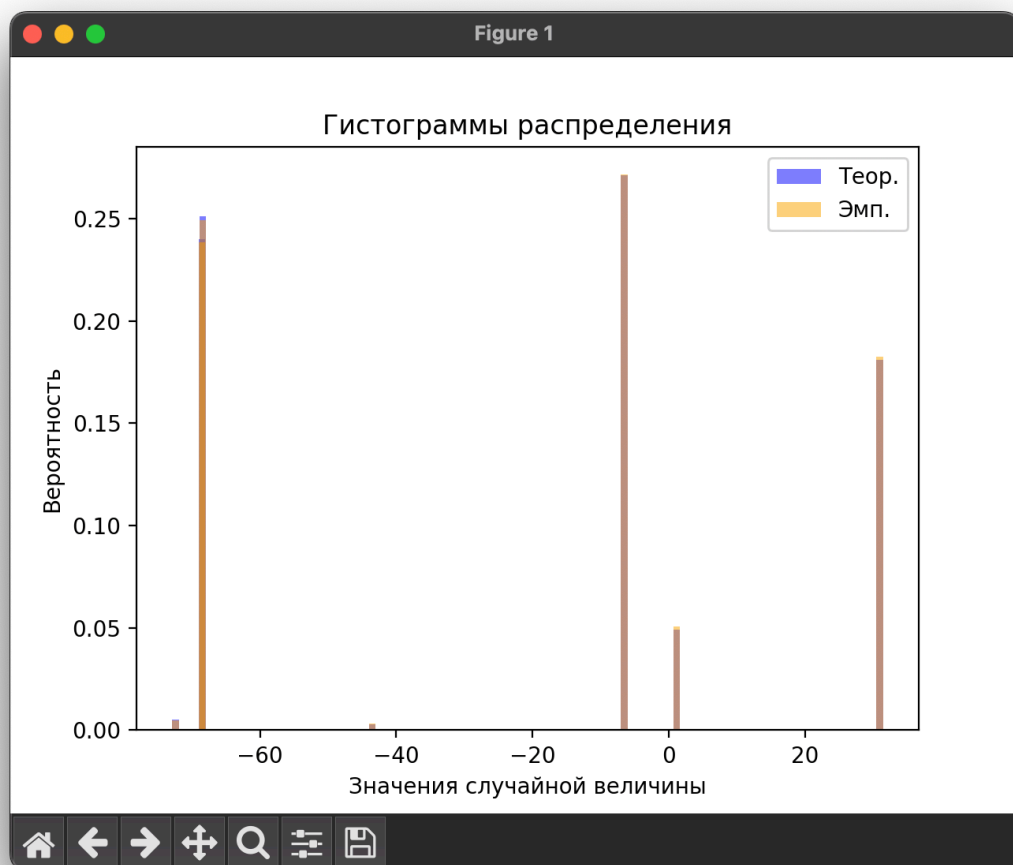


Рисунок 2.2 - Гистограммы распределения

Также были рассчитаны теоретические и фактические значения математического ожидания  $M$  и дисперсии  $D$ . Результаты представлены ниже.

Таблица 2.1 - значения  $M$  и  $D$

	$M$	$D$
Теоретическая	-30.1926	1606.2
Эмперическая	-30.0710	1600.5

### 2.3 Анализ

В результате сравнения эмпирических и теоретических значений математического ожидания и дисперсии было установлено, что они соответствуют друг другу с высокой степенью точности. Кроме того, были построены гистограммы, которые демонстрируют высокую степень совпадения с эта-

лонными значением.

Также обратим внимание на то, что всего было взято 500 точек, и даже такое маленькое расхождение в полученных значениях будет нивелировано с увеличением числа точек.

### **3 Вывод**

В ходе выполнения данной работы было реализовано программное моделирование ДСВ с использованием `FishGenerator`.

Были выполнены необходимые расчеты и построены гистограммы распределения эмпирических и теоретических вероятностей дискретной СВ. В ходе сравнительной оценки распределений эмпирических и теоретических вероятностей дискретной СВ  $x$  было установлено, что они соответствуют друг другу с высокой степенью точности.

В результате выполнения работы были получены навыки программной реализации генератора дискретной случайной величины, а также навыки анализа и сравнения эмпирических и теоретических значений математического ожидания и дисперсии. Таким образом, цель работы была достигнута.

## ПРИЛОЖЕНИЕ

```
1 from collections import Counter
2 import matplotlib.pyplot as plt
3 from fish_generator import FishGenerator
4
5
6 def calculate_properties(items):
7     mode = sum([p * x for p, x in items])
8     dispersion = sum([p * (x**2) for p, x in items]) - mode
9         ** 2
10    print(f"Мода = {mode}")
11    print(f"Дисперсия = {dispersion}")
12
13 edges = {
14     0.005: -72.4,
15     0.240: -68.5,
16     0.251: -68.4,
17     0.003: -43.5,
18     0.271: -6.5,
19     0.049: 1.2,
20     0.181: 31.0,
21 }
22
23
24 def match_edges(num):
25     for p, x in edges.items():
26         num = num - p
27         if num <= 0:
28             return x
29     return 0
30
31
32
33 num_generator = FishGenerator().generate()
34 generated_values = [match_edges(next(num_generator)[0]) for _
35                     in range(500)]
36
37 print(f"Первые 30 значений: {generated_values[:30]}\n")
38
39 value_counts = Counter(generated_values)
40 total_values = len(generated_values)
```



```

39 generated_prob = dict(
40     (count / total_values , x)
41     for x, count in value_counts.items()
42 )
43
44 print("=== Теоретические ===")
45 calculate_properties(edges.items())
46 print()
47 print("=== Эмперические ===")
48 calculate_properties(generated_prob.items())
49
50
51 # Draw graphics
52 plt.bar(edges.values(), edges.keys(),
53         width=1, color='blue', alpha=0.5, label='Теор.')
54 plt.bar(generated_prob.values(), generated_prob.keys(),
55         width=1, color='orange', alpha=0.5, label='Эмп.')
56
57 plt.xlabel('Значения случайной величины')
58 plt.ylabel('Вероятность')
59 plt.title('Гистограммы распределения')
60 plt.legend()
61 plt.show()

```

```

1 import random
2
3
4 class FishGenerator:
5     def __init__(self, seed=None):
6         self.A = [random.randint(0, 2**32 - 1) for _ in range
7                     (55)]
8         self.B = [random.randint(0, 2**32 - 1) for _ in range
9                     (52)]
10        self.index_A = 0
11        self.index_B = 0
12        random.seed(seed)
13
14    def generate(self):
15        while True:
16            self.A[self.index_A] = (
17                self.A[(self.index_A - 55) % 55] + self.A[(
18                    self.index_A - 24) % 55]) % (2**32)

```

```
16         self.B[self.index_B] = (  
17             self.B[(self.index_B - 52) % 52] + self.B[(  
                self.index_B - 19) % 52]) % (2**32)  
18         if self.B[self.index_B] & 1:  
19             yield self.A[self.index_A] / (2**32), self.B[  
                self.index_B] / (2**32)  
20         self.index_A = (self.index_A + 1) % 55  
21         self.index_B = (self.index_B + 1) % 52
```