

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение
высшего образования
АДЫГЕЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
Инженерно-физический факультет
Кафедра автоматизированных систем обработки информации и
управления

ОТЧЕТ ПО ПРАКТИКЕ

Генератор случайных чисел Парка-Миллера с
перетасовкой и без.

2 курс, группа 2УТС

Выполнил:

_____ Е. А. Ломов
«___» _____ 2024 г.

Руководитель:

_____ С. В. Теплоухов
«___» _____ 2024 г.

Майкоп, 2024 г.

1. Введение

- 1) Задание
- 2) Код прилагающий к заданию
- 3) Скриншот программы

Содержание

1. Введение	2
2. Теория	3
2.1. Техническое задание	3
2.2. Теоретическая часть	3
3. Ход работы	3
3.1. Код прилагающий к заданию	3

2. Теория

2.1. Техническое задание

Задание: Генератор случайных чисел Парка-Миллера с перетасовкой и без.

2.2. Теоретическая часть

Самая простая последовательность, которую можно предложить для реализации генератора равномерного распределения:

$$I(j+1)=a \cdot I(j) \pmod{m}$$

при соответствующем выборе констант. Константы были предложены Park и Miller:

$$a=7^5=16807, m=2^{31}-1=2147483647.$$

Модуль разлагается в выражение:

$$m=a \cdot q + r$$

Если $r < q$ и $0 < z < m-1$, то при этом величины $a \cdot (z \bmod q)$ и $r \cdot [z/q]$ всегда лежат в интервале $0, \dots, m-1$. Для умножения $(a \cdot z) \pmod{m}$ при этом используется алгоритм:

- $t = a(z \bmod q) - r[z/q]$
- если $t < 0$, то $t += m$.
- $(a \cdot z) \pmod{m} = t$.

В случае констант Парка-Миллера можно использовать $q=12773$ и $r=2836$.

3. Ход работы

3.1. Код прилагающий к заданию

```
#Генератор случайных чисел Парка-Миллера с перетасовкой и без.
class ParkMillerRandom:
    def __init__(self, seed, shuffle=False):
        self.seed = seed
        self.shuffle = shuffle
        self.a = 16807
        self.m = 2147483647
```

```

        self.random_numbers = []

    def generate(self, n):
        if self.shuffle:
            self.random_numbers = []

        x = self.seed
        for _ in range(n):
            x = (self.a * x) % self.m
            if self.shuffle:
                self.random_numbers.append(x)
            yield x

    def get_random_numbers(self):
        return self.random_numbers

# Пример использования
seed = 232312213
n = 10

# Генератор без перетасовки
random_gen = ParkMillerRandom(seed)
random_numbers = list(random_gen.generate(n))
print(f'Случайные числа без перетасовки: {random_numbers}')

# Генератор с перетасовкой
random_gen_shuffle = ParkMillerRandom(seed, shuffle=True)
list(random_gen_shuffle.generate(n))
random_numbers_shuffle = random_gen_shuffle.get_random_numbers()
print(f'Случайные числа с перетасовкой: {random_numbers_shuffle}')

```

```
C:\Users\sync\anaconda3\python.exe
Random numbers without shuffling: [207482415, 1790989824, 2035175616, 77048696, 24794531, 109854999, 1644515420, 1256127050, 1963079340, 1683198519]
Random numbers with shuffling: [207482415, 1790989824, 2035175616, 77048696, 24794531, 109854999, 1644515420, 1256127050, 1963079340, 1683198519]
Press any key to continue . . .
```

Рис.1 Скриншот программы