



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ _____ «Информатика и системы управления»

КАФЕДРА _____ «Программное обеспечение ЭВМ и информационные технологии»

ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ по курсу «Моделирование»

«Генерация случайных чисел»

| | | | |
|----------------|----------------------------|--------------------------|---|
| Студент: | <u>ИУ7-73Б</u> (группа) | _____ (подпись, дата) | <u>М. Д. Маслова</u> (И. О. Фамилия) |
| Преподаватель: | | _____ (подпись, дата) | <u>И. В. Рудаков</u> (И. О. Фамилия) |

2022 г.

СОДЕРЖАНИЕ

| | | |
|----------|---|----------|
| 1 | Задание | 4 |
| 2 | Теоретическая часть | 5 |
| 2.1 | Методы получения последовательности случайных чисел | 5 |
| 2.1.1 | Алгоритмический способ | 5 |
| 2.1.2 | Табличный способ | 5 |
| 2.2 | Критерий случайности | 6 |
| 3 | Практическая часть | 7 |
| 3.1 | Текст программы | 7 |
| 3.2 | Полученный результат | 9 |

1 Задание

Разработать программное обеспечение, предоставляющее возможность генерации последовательности случайных чисел алгоритмическим и табличным способом, а также возможность расчета коэффициента критерия случайности по полученным последовательностям.

Реализовать графический интерфейс, позволяющий пользователю ввести последовательность для проверки ее случайности.

2 Теоретическая часть

2.1 Методы получения последовательности случайных чисел

Для генерации случайных чисел применяются следующие способы:

- *аппаратный*, в основе которого лежат физические эффекты;
- *табличный*, при использовании которого заранее полученные и проверенные случайные числа оформлены в виде таблицы в памяти ЭВМ;
- *алгоритмический*, с помощью которого формируются детерминированные последовательности чисел, где каждое число зависит от предыдущего, но для стороннего наблюдателя такие последовательности выглядят случайными, из-за чего называются псевдослучайными.

2.1.1 Алгоритмический способ

В данной работе реализуется *квадратичный конгруэнтный метод*, в котором последовательность чисел формируется следующим образом:

$$y_{n+1} = (Ay_n^2 + By_n + C) \bmod m, \quad (2.1)$$

где $m = 2^l$. Если $l \geq 2$, то наибольшее значение составляет $T_{\max} = 2^l$, что достигается при четном A , нечетном C и если нечетное B удовлетворяется условию $B \bmod 4 = (A + 1) \bmod 4$.

2.1.2 Табличный способ

В данной работе для генерации случайных чисел табличным способом используются цифры из части таблицы «*A Million Random Digits with 100,000 Normal Deviates*», опубликованной в 1955 году.

Данная таблица сохранена в виде текстового файла. Для генерации чисел выбирается начальная позиция в файле, читаются следующие n цифр, где n — количество разрядов в генерируемом числе, и из строки последовательность преобразуется в число. Для генерации следующего числа происходит переход к следующей строке таблицы с сохранением номера столбца. При невозможности перейти к следующей строке в связи с окончанием файла позиция переводится на первую строку, а номер столбца увеличивается на единицу. Если цифр в строке не хватает для формирования числа, они берутся из начала

следующей строки.

2.2 Критерий случайности

В данной работе для оценки случайности предлагается критерий автора на основе углов между векторами, координаты начала и конца которых состояются из двух соседних пар последовательности с одним общим числом. Критерий состоит в следующем.

Из элементов сгенерированной последовательности x_1, x_2, \dots, x_N длиной N формируются пары (x_i, x_{i+1}) , где $i = 1 \dots N - 1$. Далее каждая из этой пары воспринимается, как координаты точки на плоскости.

Для каждой пары координат точек $(x_i, x_{i+1}), (x_{i+1}, x_{i+2})$ ищется их соединяющий вектор \vec{v}_i с координатами $(x_{i+1} - x_i, x_{i+2} - x_{i+1})$. Из векторов также формируется последовательность.

Далее для каждого вектора происходит поиск углов между ним и k ближайших векторов в последовательности. Угол между векторами θ определяется по формуле:

$$\cos \theta = \frac{\vec{v}_i \cdot \vec{v}_j}{\|\vec{v}_i\| \|\vec{v}_j\|}. \quad (2.2)$$

Углы равные 0 и π являются признаком линейной зависимости, поэтому им сопоставляется коэффициент $K = 0$. При данном алгоритме формирования координат точек из последовательности угол величиной $\frac{\pi}{2}$ между векторами, параллельными осям координат свидетельствует о повторении чисел, однако повторение чисел в случайной последовательности приемлемо, при этом угол величиной $\frac{\pi}{2}$ может быть получен и при других положениях векторов, поэтому данному углу сопоставляется коэффициент равные $K = \frac{1}{2}$. Углам величиной $\frac{\pi}{4}$ и $\frac{3\pi}{4}$ сопоставляется коэффициент $K = 1$, как средним между названными выше углами. Коэффициенты K для остальных углов рассчитываются по формуле ($\theta' = \theta$, если $\theta \leq \frac{\pi}{2}$, $\theta' = \pi - \theta$, иначе):

$$K = \begin{cases} \frac{4\theta'}{\pi}, & \text{если } 0 \leq \theta' < \frac{\pi}{4}; \\ \frac{-2\theta'}{\pi} + \frac{3}{2}, & \text{если } \frac{\pi}{4} \leq \theta' \leq \frac{\pi}{2}, \end{cases} \quad (2.3)$$

Итоговый коэффициент лежащий в промежутке $[0, 1]$ считается как отношение суммы найденных коэффициентов K_i к количеству найденных углов.

3 Практическая часть

3.1 Текст программы

На листинге 3.1 представлена реализация квадратичного когруэнтного метода генерации случайных чисел.

Листинг 3.1 – Реализация квадратичного когруэнтного метода генерации случайных чисел

```
1 class Generator:
2
3     def __init__(self, normGen, lower=0, upper=100):
4         self.normGen = normGen
5         self.lower = lower
6         self.upper = upper
7
8     def GenerateNumber(self):
9         return round(self.normGen.GetNumber01()
10                     * (self.upper - self.lower) + self.lower)
11
12     def GenerateSequence(self, length):
13         return [self.GenerateNumber() for _ in range(length)]
14
15
16 class QuadraticGenerator(Generator):
17
18     def __init__(self, lower, upper):
19         super().__init__(QuadraticRandom(), lower, upper)
20
21
22 class QuadraticRandom(NormalizedRandom):
23
24     def __init__(self):
25         self.current = 4001
26         self.A = 6
27         self.B = 7
28         self.C = 3
29         self.m = 8192
30
31     def GetNumber01(self):
32         self.current = (self.A * self.current * self.current
33                       + self.B * self.current
34                       + self.C) % self.m
35         return self.current / self.m
```

На листинге 3.2 представлена реализация табличного способа получения последовательности случайных чисел.

Листинг 3.2 – Реализация табличного способа генерации случайных чисел

```
1 from datetime import datetime
2
3 PAGES_NUMBER = 7
4 ROWS_PER_PAGE = 50
5 COLS_PER_ROW = 50
6
7 SYMBOLS_PER_PAGE = ROWS_PER_PAGE * COLS_PER_ROW
8 SYMBOLS_NUMBER = SYMBOLS_PER_PAGE * PAGES_NUMBER
9
10 class TabularGenerator:
11
12     def __init__(self):
13         page = datetime.now().microsecond % PAGES_NUMBER
14         row = datetime.now().microsecond % ROWS_PER_PAGE
15         column = datetime.now().microsecond % COLS_PER_ROW
16
17         self.position = SYMBOLS_PER_PAGE * page + COLS_PER_ROW *
            row + column
18
19
20     def GenerateNumber(self, digits=1):
21         num = -1
22         with open("randseq/data/digits.txt", "r") as f:
23             notRead = True
24
25             while notRead:
26                 f.seek(self.position, 0)
27                 num = int(f.read(digits))
28
29                 if num // (10 ** (digits - 1)) >= 1:
30                     notRead = False
31
32                 self.position += COLS_PER_ROW
33
34                 if self.position > SYMBOLS_NUMBER - 1:
35                     self.position %= SYMBOLS_NUMBER
36                     self.position += 1
37
38     return num
```

На листинге 3.3 представлена функция вычисления коэффициента описанного выше критерия оценки случайности.

Листинг 3.3 – Реализация функции расчета коэффициента критерия оценки случайности

```
1 class RandomnessCriterion:
2
3     def vectorAngle(self, vector1, vector2):
4         dotProduct = vector1.dot(vector2)
5         normsProduct = np.linalg.norm(vector1) * np.linalg.norm(
6             vector2)
7
8         return np.rad2deg(np.arccos(dotProduct / normsProduct))
9
10    def GetCoefficient(self, sequence):
11        points = [np.array([x, y]) for x, y in zip(sequence[:-1],
12            sequence[1:])]
13
14        vectors = [points[i + 1] - point for i, point in
15            enumerate(points[:-1])]
16
17        angles = []
18        score = 0
19        for i, vector1 in enumerate(vectors[:-1]):
20            end = i + 11
21            if end > len(vectors):
22                end = len(vectors)
23
24            for j, vector2 in enumerate(vectors[i + 1:end]):
25                angle = self.vectorAngle(vector1, vector2)
26                angles.append(angle)
27
28                firstQuarterAngle = 180 - angle if angle > 90
29                else angle
30                score += (firstQuarterAngle / 45
31                    if firstQuarterAngle < 45
32                    else - firstQuarterAngle / 90 + 1.5)
33
34        return score / len(angles) if angles else 0
```

3.2 Полученный результат

На рисунке 3.1 приведен результат работы программы при генерации 5000 чисел алгоритмическим и табличным методами.

На рисунке 3.2 приведен результат расчета коэффициента критерия случайности для постоянной последовательности.

Алгоритмический способ

Количество чисел 5000 - +

| | 1 | 2 | 3 |
|----|---|----|-----|
| 1 | 0 | 13 | 126 |
| 2 | 4 | 45 | 453 |
| 3 | 1 | 17 | 170 |
| 4 | 6 | 70 | 705 |
| 5 | 6 | 68 | 685 |
| 6 | 4 | 52 | 527 |
| 7 | 3 | 38 | 383 |
| 8 | 8 | 94 | 946 |
| 9 | 3 | 36 | 363 |
| 10 | 6 | 74 | 744 |
| 11 | 2 | 29 | 290 |
| 12 | 6 | 72 | 726 |
| 13 | 5 | 64 | 647 |
| 14 | 0 | 12 | 122 |

Коэффициенты случайности

| | 1 | 2 | 3 |
|---|-------|-------|-------|
| К | 0.597 | 0.615 | 0.615 |

Сгенерировать и рассчитать

Табличный способ

Количество чисел 5000 - +

| | 1 | 2 | 3 |
|----|---|----|-----|
| 1 | 2 | 75 | 327 |
| 2 | 9 | 97 | 718 |
| 3 | 2 | 39 | 780 |
| 4 | 2 | 27 | 690 |
| 5 | 9 | 31 | 398 |
| 6 | 2 | 43 | 335 |
| 7 | 5 | 34 | 554 |
| 8 | 5 | 24 | 293 |
| 9 | 8 | 84 | 385 |
| 10 | 5 | 85 | 802 |
| 11 | 2 | 49 | 988 |
| 12 | 6 | 84 | 646 |
| 13 | 1 | 25 | 613 |
| 14 | 5 | 95 | 644 |

Коэффициенты случайности

| | 1 | 2 | 3 |
|---|-------|-------|-------|
| К | 0.592 | 0.611 | 0.613 |

Сгенерировать и рассчитать

Рисунок 3.1 – Пример работы на последовательности из 5000 чисел

Ввод последовательности

| | | | 1 | |
|----|---|---|---|--|
| 1 | 1 | - | + | |
| 2 | 1 | - | + | |
| 3 | 1 | - | + | |
| 4 | 1 | - | + | |
| 5 | 1 | - | + | |
| 6 | 1 | - | + | |
| 7 | 1 | - | + | |
| 8 | 1 | - | + | |
| 9 | 1 | - | + | |
| 10 | 1 | - | + | |

Коэффициент случайности

0.00

-

+

Рассчитать

Рисунок 3.2 – Пример работы на постоянной последовательности

На рисунке 3.3 приведен результат расчета коэффициента критерия случайности для возрастающей последовательности.

На рисунке 3.4 приведен результат расчета коэффициента критерия слу-

Ввод последовательности

| 1 | | |
|----|---|-----|
| 1 | 0 | - + |
| 2 | 1 | - + |
| 3 | 2 | - + |
| 4 | 3 | - + |
| 5 | 4 | - + |
| 6 | 5 | - + |
| 7 | 6 | - + |
| 8 | 7 | - + |
| 9 | 8 | - + |
| 10 | 9 | - + |

Коэффициент случайности

0.00 - +

Рассчитать

Рисунок 3.3 – Пример работы на возрастающей последовательности

чайности для строго убывающей последовательности.

Ввод последовательности

| 1 | | |
|----|---|-----|
| 1 | 9 | - + |
| 2 | 8 | - + |
| 3 | 7 | - + |
| 4 | 6 | - + |
| 5 | 5 | - + |
| 6 | 4 | - + |
| 7 | 3 | - + |
| 8 | 2 | - + |
| 9 | 1 | - + |
| 10 | 0 | - + |

Коэффициент случайности

0.00 - +

Рассчитать

Рисунок 3.4 – Пример работы на убывающей последовательности

На рисунке 3.5 приведен результат расчета коэффициента критерия случайности для периодической последовательности.

Ввод последовательности

| 1 | | |
|----|---|-----|
| 1 | 1 | - + |
| 2 | 9 | - + |
| 3 | 1 | - + |
| 4 | 9 | - + |
| 5 | 1 | - + |
| 6 | 9 | - + |
| 7 | 1 | - + |
| 8 | 9 | - + |
| 9 | 1 | - + |
| 10 | 9 | - + |

Коэффициент случайности

| | | |
|------|---|---|
| 0.00 | - | + |
|------|---|---|

Рассчитать

Рисунок 3.5 – Пример работы на периодической последовательности

На рисунке 3.6 приведен результат расчета коэффициента критерия случайности для случайной последовательности из 10 чисел.

Ввод последовательности

| 1 | | |
|----|---|-----|
| 1 | 3 | - + |
| 2 | 8 | - + |
| 3 | 4 | - + |
| 4 | 7 | - + |
| 5 | 2 | - + |
| 6 | 5 | - + |
| 7 | 0 | - + |
| 8 | 6 | - + |
| 9 | 9 | - + |
| 10 | 1 | - + |

Коэффициент случайности

| | | |
|------|---|---|
| 0.46 | - | + |
|------|---|---|

Рассчитать

Рисунок 3.6 – Пример работы на случайной последовательности