**Лабораторная работа 2**

Тема: "Имитационное моделирование. Моделирование частотной характеристики. Моделирование случайных событий"

Цели работы:

1. изучение методов частотного анализа дискретных сигналов с помощью дискретного преобразования Фурье (MS Excel| Google sheets и python);
2. изучение методов и алгоритмов моделирования случайных чисел (MS Excel | Google sheets и python).

Часть 2. Моделирование случайных чисел

Содержание

[Задача 1. Дискретное преобразование Фурье 1](#_Toc96075452)

[Варианты задания 1](#_Toc96075453)

[Примечания 2](#_Toc96075454)

[Теоретические данные 2](#_Toc96075455)

[Пример выполнения работы 5](#_Toc96075456)

# Задача 2. Моделирование случайных чисел

Согласно варианту задания необходимо составить и отладить программу (подпрограмму) генерирования случайных чисел:

1. с равновероятным распределением на интервале [0; 1), используя формулу
2. используя готовый пакет Google | Excel надстройку Анализ данных и библиотеку numpy алгоритм из объекта random.

Провести статистическое исследование генератора при различных значениях выборки: малых n<25, средних , больших n>500.

Вариант задания выбирается из табл. 1, в которой указаны тип генератора случайных чисел, начальные условия.

Отчет по заданию должен содержать:

* Описание метода генерирования псевдослучайных чисел;
* Программу (подпрограмму) датчика псевдослучайных чисел;
* Вывод 20 случайных чисел;
* Результаты статистической обработки при различных значениях выборки;
* \* диаграмма вывода ГСЧ;
* Выводы по задаче – сравнение и оценка генераторов.

# Варианты задания

Таблица 1

Тип генератора случайных чисел и начальные условия

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Вариант** | **Тип генератора** | **Начальные данные** | **Метод в numpy**  **(тип распределения в numpy.random)** | **Метод в Google|Excel**  **(надстройка ГСЧ)** |
| 1 | Мультипликативный конгруэнтный метод.  Формула в теории (1). | *k*=31; *а*=16807 | Гамбала | Равномерное |
| 2 | Линейный конгруэнтный метод. Формула в теории (3). | *а*=16070093; *b*=453816693; *k*=31 | Бета- распределение | Нормальное |
| 3 | Квадратичный конгруэнтный метод.  Формула в теории (4). | *а*=6; *b*=7; *c*=3; *m*=4096 | биномиальное | Бернулли |
| 4 | Кубический конгруэнтный метод.  Формула в теории (5). | *а*=53668; *b*=400144; *c*=12211; *k*=15 | хи-квадрат | Биноминальное |
| 5 | Генератор Фибоначчи.  Формула в теории (6). | *k*=14 | Дирихле | Пуассона |
| 6 | BBS – генератор.  Формула в теории (7). |  | экспоненциальное | Модельное |
| 7 | Системный генератор MS Fortran.  Формула в теории (8). | - | Фишера | Дискретное |
| 8 | Тригонометрический метод. Формула в теории (9). | - | Гамма | Равномерное |
| 9 | Тригонометрический метод со счетчиком.  Формула в теории (10). | - | геометрическое | Нормальное |
| 10 | Мультипликативный конгруэнтный метод.  Формула в теории (1). | *k*=15; *а*=16807 | Лапласа | Бернулли |
| 11 | Линейный конгруэнтный метод. Формула в теории (3). | *а*=165; *b*=3463; *k*=14 | логистическое | Биноминальное |
| 12 | Квадратичный конгруэнтный метод.  Формула в теории (4). | *а*=8; *b*=9; *c*=3; *m*=4096 | логарифмическое | Пуассона |
| 13 | Кубический конгруэнтный метод.  Формула в теории (5). | *а*=52774; *b*=40692; *c*=3791; *k*=31 | мультиномиальное | Модельное |
| 14 | Генератор Фибоначчи.  Формула в теории (6). | *k*=14 | нецентральное Фишера | Дискретное |
| 15 | BBS – генератор.  Формула в теории (7). |  | нормальное (Гаусса) | Равномерное |
| 16 | Системный генератор MS Fortran.  Формула в теории (8). | - | Парето | Нормальное |
| 17 | Тригонометрический метод. Формула в теории (9). | - | Рэлея | Бернулли |
| 18 | Тригонометрический метод со счетчиком.  Формула в теории (10). | - | Коши | Биноминальное |
| 19 | Мультипликативный конгруэнтный метод.  Формула в теории (1). | *k*=31; *а*=16807 | Стьюдента | Пуассона |
| 20 | Линейный конгруэнтный метод. Формула в теории (3). | *а*=16070093; *b*=453816693; *k*=31 | Вальда | Модельное |

К заданию в elearning прикрепляется решение задания:

1. архив с файлами \*.xlsx и папкой с \*.py

или

1. ссылка на открытый Google Sheets документ и программу на диске в файле \*.txt.

# Примечания

1. Предварительно установить надстройку Анализ данных (генератор случайныйх чисел) в MS Excel или расширение XLMiner Analysys ToolPak (Random number generation) в Google
2. Предварительно установить python версии 3.\* (3.7.5 или 3.8, например) и библиотеки matplotlib (или plotly), numpy через pip install. Решение может быть сделано в Google Colab, там эти библиотеки есть.

# Теоретические данные

## ***Общие сведения***

Моделирование на ЭВМ процессов функционирования различных систем связано с выработкой большого количества случайных чисел с заданными законами распределения. Для этой цели используется обычно один из следующих способов:

- *табличный* (файловый) – ввод таблиц равномерно распределённых случайных чисел во внешнюю или оперативную память ЭВМ;

- *аппаратный* (физический) – использование специального приспособления к ЭВМ – "датчика" случайных чисел, формирующего случайные величины путём физического моделирования некоторых случайных процессов (излучения радиоактивных источников, шумов электронных ламп и др.);

- *алгоритмический* (программный) – использование псевдослучайных (квазислучайных) последовательностей, реализуемых программным генератором случайных чисел.

*Псевдослучайными* числами называются числа, вырабатываемые ЭВМ рекуррентным способом по специальным алгоритмам, когда каждое последующее число  получается из предыдущих в результате применения некоторых арифметических и логических операций. Такая последовательность чисел удовлетворяет известным критериям случайности, хотя входящие в эту последовательность числа зависимы между собой. Одним из недостатков этого метода является периодичность образованных программным способом псевдослучайных чисел, но для ряда задач, не требующих большого количества случайных чисел, длина периода является достаточной.

Достоинства метода псевдослучайных чисел.

1. На получение каждого случайного числа затрачивается несколько простых операций, так что скорость генерирования случайных чисел имеет тот же порядок, что и скорость работы ЭВМ.
2. Малый объем памяти ЭВМ для программирования.
3. Любое из чисел легко воспроизвести.
4. Качество генерируемых случайных чисел достаточно проверить один раз.

## ***Генераторы случайных чисел***

***Метод Неймана.*** Первый алгоритм для получения псевдослучайных чисел был предложен Дж. Нейманом. Его называют методом середины квадратов.

Пусть задано четырехзначное число . Возведем его в квадрат. Получим 8-значное число . Выберем 4 средние цифры из этого числа и положим . Затем снова возведем в квадрат и извлечем из него 4 средние цифры. Получим и т.д.

**а) конгруэнтные методы**

Самое широкое применение при моделировании на ЭВМ получили конгруэнтные методы генерации псевдослучайных последовательностей, в основе которых лежит фундаментальное понятие *конгруэнтности*.

Два целых числа *А* и *В* конгруэнтны (сравнимы) по модулю *m*, если их разность делится на *m* без остатка. То есть

*В* - *А* = *km* ,

где *k* - целое. Это определение записывается так *A = B* (mod *m*).

*Например*:

1984 ≡ 4 (mod10), 5008 ≡ 8 (mod 103) и т.д.

Величина *m* берется равной длине машинного слова *m* = 2*b*, где *b* — число бит в машинном слове.

***Мультипликативный конгруэнтный метод (АлгоритмЛемера).***

Последовательность случайных чисел  получается с помощью следующего итерационного соотношения:

, (1)

где  и  - очередное и последующее случайное число;

*m* - модуль, *m* > 0;

*a*  -  множитель, 0 ≤ *a*< *m*;

 - начальное значение,  .

Если *m* и *а* являются целыми, то создается последовательность целых чисел в диапазоне . Из рекуррентной природы формулы следует, что при одном и том же значении  мы получим при повторной генерации ту же самую последовательность.

Последовательность случайных чисел периодически повторяются. Это связано с тем, что числа  могут принимать только значения . То есть, самое большее через  шагов уже один раз полученное число должно появиться опять, а с ним повторяется и вся последовательность. Таким образом, длина периода при модуле  не может превышать . Поэтому выбор значений *а* и *m* является критичным для разработки хорошего генератора случайных чисел. Для практических расчетов принимают .

Последовательность чисел , равномерно распределенных на интервале от нуля до единицы, рассчитывается по формуле

. (2)

***Линейный конгруэнтный метод***. Схема метода предложена Д.Лехмером в 1949 году. Работа этих генераторов основана на использовании формулы:

. (3)

Число *m* выбирается аналогично предыдущему пункту. Множитель *а* предпочтительно выбирать в интервале [0,01*m*; 0,99*m*]. Значение *с* может быть произвольным, но не должно иметь общего множителя с *m*. При *b*=0 получаем, рассмотренный выше мультипликативный метод.

С вычислительной точки зрения линейный метод генерации последовательности неотрицательных целых чисел сложнее мультипликативного на одну операцию сложения, но при этом возможность выбора дополнительного параметра позволяет уменьшить возможную корреляцию получаемых чисел. Однако экспериментальная проверка качества генерируемой последовательности чисел на основе этой формулы является сложнее.

***Квадратичный конгруэнтный метод***. Последовательность случайных чисел вычисляется с помощью следующего рекуррентного соотношения

. (4)

Наибольшее значение периода данного датчика достигается при четном *a*, нечетном *с* и если нечетное *b* удовлетворяет условию .

***Кубический конгруэнтный метод.*** Последовательность случайных чисел вычисляется с помощью следующего рекуррентного соотношения

 (5)

Для практических расчетов принимают .

***Генератор Фибоначчи.*** Последовательность случайных чисел вычисляется с помощью следующего рекуррентного соотношения

 (6)

где  и  - целые числа, лежащие между нулем и *m* .

***BBS – генератор.*** Данный генератор вычисляется по формуле

. (7)

Вначале выбираются два больших простых числа *p* и *q.* Числа *p* и *q* должны удовлетворять условиям , т.е. при делении *p* и *q* на 4 должен получиться одинаковый остаток 3. Далее вычисляется число , называемое целым числом Блюма. По формуле вычисляется стартовое число генератора, где - взаимно простое число (т.е. не имеющее общих делителей, кроме 1).

***Системный генератор MS Fortran.***

. (8)

**б) тригонометрические методы**

Основаны на использовании ошибки вычисления косинуса больших алгоритмов.

, (9)

где *n* - целое число, определяемое типом ЭВМ;

 - начальное значение,  .

***Тригонометрический метод со счетчиком***

, (10)

где  - начальное значение,  .

## ***Проверка качества работы генератора случайных чисел (ГСЧ)***

Применяемые генераторы случайных чисел перед моделированием должны пройти тщательное предварительное тестирование на равномерность, стохастичность и независимость получаемых последовательностей случайных чисел. Существует множество статистических критериев, которые можно использовать для проверки того, будет ли последовательность случайной. Наиболее точным считается спектральный критерий. Например, очень распространенный критерий, называемый *КС*-критерием, или критерием Колмогорова - Смирнова.

Можно воспользоваться приближенным тестом, который состоит в проверке на равномерность распределения *N* случайных чисел .

а) *проверка статистических характеристик*. Тест состоит в вычислении математического ожидания  и дисперсии  полученных случайных чисел. Согласно этому тесту, для равномерного распределения должны выполняться условия:

;

;

.

б) *частотный тест*. Позволяет выяснить, сколько чисел попало в интервал , то есть (0.5 – 0.2887; 0.5 + 0.2887) или, в конечном итоге, (0.2113; 0.7887). Так как 0.7887 – 0.2113 = 0.5774, заключаем, что в хорошем ГСЧ в этот интервал должно попадать около 57,7% из всех выпавших случайных чисел (см. **рис.  1**).

|  |  |
| --- | --- |
| |  | | --- | |  | |

**Рис. 1. Частотная диаграмма идеального ГСЧ  
в случае проверки его на частотный тест**

Также необходимо учитывать, что количество чисел, попавших в интервал (0; 0.5), должно быть примерно равно количеству чисел, попавших в интервал (0.5; 1).

# Подробнее

1. <https://habr.com/ru/post/415373/>
2. <https://exceltable.com/funkcii-excel/generator-sluchaynyh-chisel>
3. <https://numpy.org/doc/stable/reference/random/index.html>
4. <https://blog.sheetgo.com/finance-processes/how-to-use-xlminer-analysis-toolpak-add-on-for-google-sheets/>