МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

федеральное государственное автономное
образовательное учреждение высшего образования
«Самарский национальный исследовательский университет
имени академика С.П. Королева»
(Самарский университет)

Институт информатики и кибернетики

Кафедра информационных систем и технологий

На правах рукописи

УДК 004

Тышкун Андрей Юрьевич

АВТОМАТИЗИРОВАННАЯ СИСТЕМА ИССЛЕДОВАНИЯ КАЧЕСТВА ГЕНЕРАТОРОВ СЛУЧАЙНЫХ И ПСЕВДОСЛУЧАЙНЫХ ЧИСЕЛ

Автореферат
выпускной квалификационной работы магистра
по направлению подготовки 09.04.01 «Информатика и вычислительная техника»
профиль «Информационные системы и технологии»

Самара – 2023 год

Работа выполнена в Самарском университете на кафедре информационных систем и технологий.

Научный руководитель: к.т.н., доцент Климентьев К.Е.

Рецензент: Никитин Константин Александрович, доцент кафедры информационных систем и технологий ФГБОУ ВО ПГУТИ, к.т.н.

ОБЩАЯ ХАРАКТЕРИСТИКА РАБОТЫ

Актуальность темы

В современном мире большое значение имеют генераторы случайных и псевдослучайных последовательностей бит. Они широко применяются в различных задачах моделирования, например, методы Монте-Карло полагаются на повторяющуюся случайную выборку для получения численных результатов, в статистической выборке, азартных играх, для рандомизации дизайна, для генерации паролей пластиковых карт и других областях, в которых желательно получение непредсказуемых результатов. Случайные и псевдослучайные последовательности имеют огромную роль для криптографии – от их качества зависит секретность информации. Практически не найти технологии, в которой не были бы нужны случайные числа, например, для генерации сессионных ключей для безопасной сетевой сессии, используемые в таких технологиях как SSL/TLS, VPN, для установки простой TCP-сессии – два раза генерируются случайные числа для «handshake», для защиты DNS, чтобы не подделали ответы – генерируется много «salt».

Развитие ЭВМ, с одной стороны, расширило круг задач, использующих случайные числа, а с другой – предъявило высокие требования к качеству их генерации. Поэтому оценка эффективности генераторов случайных чисел представляет большой интерес. При реализации генераторов случайных чисел принципиально важно определить, насколько последовательность чисел или бит, сгенерированных генератором, является случайной, это позволяет быть уверенным, что порождаемые числа, действительно имеют случайную природу и могут быть использованы для вышеупомянутых областей применения. Как правило, для этой цели используются различные тесты, которые способны распознать определенные закономерности в исследуемых числовых последовательностях, с помощью сравнения их характеристик с аналогичными характеристиками истинно случайной последовательности. Данная работа как раз посвящена созданию системы для тестирования качества генераторов случайных и псевдослучайных чисел.

Разработка данной системы является актуальной и по сей день, несмотря на множество уже имеющихся аналогичных средств тестирования генераторов, например, такие системы как Diehard или TestU01, не имеют интуитивно понятного интерфейса для пользователя и разработаны только под операционную систему Linux, также многие системы являются не универсальными, имеют жестко заданные параметры тестирования, в результате система может просто

не поддерживать анализ определенного типа не случайности, требуемый пользователю, или поддерживать, но не применительно к заданным параметрам генератора, для неспециалиста некоторые системы малоинформативные, используют сторонние библиотеки, во многих отсутствует графическое тестирование, медленные, например, testu01.crush работает больше часа, testu01.bigcrush несколько часов, к тому же там есть ошибки. Краткие результаты приведены в таблице 1.

Целью работы является создание программного обеспечения, обладающего возможностью проверки гипотезы о истинной случайности последовательностей чисел, генерируемых различными генераторами, и выявление их различных отклонений от случайности, если такое присутствует, с помощью универсальной батареи тестов, включающей в себя различные статистические и графические тесты.

Таблица 1 – Сравнительный анализ программных продуктов

Хар-ка/ Продукт	DIEHARD	TestU01	NIST SP 800-22	Разработанная система
Качественные тесты	+	+	+	+
Встроенные генераторы	+	+	+	+
Битовые тесты	-	-	+	+
Скорость	-	-	+	+
Универсальность	-	-	-	+
Графические тесты	-	-	-	+
Переносимость	-	-	+	+
Интуитивно понятный интерфейс	-	-	-	+
Русский язык	-	-	+	+
База данных тестирования	-	-	-	+
Эмпирические критерии	+	+	+	+
Теоретические критерии	-	-	-	+
Градация тестов по их силе	-	+	-	+
Определение области тестирования	-	-	+	+
Настройки параметров тестов/тестирования	-	+	-	+
Возможность получения интегральной оценки	-	+	+	+
Возможность тестирования нескольких последовательностей	-	+	+	+
Отчет по каждому тесту	+	+	+	+

В соответствии с поставленной целью в выпускной квалификационной работе магистра решаются следующие задачи исследования:

1. Анализ генераторов случайных и псевдослучайных чисел.
2. Анализ способов выявления неслучайности в генерируемых последовательностях.
3. Обзор существующих систем аналогов.
4. Разработка информационно-логической модели системы.
5. Разработка, реализация и отладка программного обеспечения.
6. Исследование генераторов реализованной системой.

Методы исследования, используемые в выпускной квалификационной работе магистра, основаны на положениях теории вероятностей и математической статистики, теории оптимизации.

Научная новизна работы заключается в исследовании свойств различных типов генераторов псевдослучайных случайных чисел в зависимости от параметров тестирования, благодаря универсальной системе тестирования генераторов.

Практическая ценность работы заключается в разработке алгоритмического и программного обеспечения автоматизированной системы, позволяющего решать следующие задачи:

1. Генерация и загрузка последовательности чисел для их тестирования.
2. Выбор алгоритмов и параметров для тестирования.
3. Исследование последовательностей на случайность или нахождение отклонений от нее.
4. Предоставление оценки о свойствах исследуемой последовательности.

Положения, выносимые на защиту:

1. Исследование свойств различных генераторов случайных чисел.
2. Автоматизированное тестирование генерируемых последовательностей, позволяющее использовать все функции системы: загрузка и формирование данных, настройка параметров тестирования и тестов, быстрое тестирование различными качественными тестами, представление результатов тестирования.

СОДЕРЖАНИЕ РАБОТЫ

Во введении приведены основные определения и понятия, показана актуальность темы выпускной квалификационной работы.

В первой главе приведена постановка задачи, определение случайных чисел, представлены различные генераторы случайных чисел, осуществлен анализ существующих систем, использующихся для решения задачи тестирования генераторов, проведен анализ способов тестирования генераторов случайных чисел и оценки их результатов, описаны выбранные методы решения поставленной задачи.

Для создания системы, которая будет исследовать генераторы случайных и псевдослучайных чисел, существует не так много методологических подходов, которые дают рекомендации по созданию «качественной» системы тестирования, которая должна удовлетворять следующим условиям:

1. Тесты набора должны находить все возможные виды отклонений от случайности и находить скрытые зависимости.
2. Тесты должны быть независимыми. Не следует использовать тестов больше, чем это необходимо.
3. Тесты должны быть универсальными. Многие описанные тесты рассчитаны на определенный вид последовательностей.

Поэтому проведя анализ существующих тестов, были выбраны следующие статистические и графические тесты, которые удовлетворяют всем описанным выше критериям, представлены в таблице 2.

Таблица 2 – Подобранные тесты

Название теста	Описание
Статистические тесты	
1	2
Тест серий	Проверяет неравномерность распределения m-битных слов.
Проверка гипотезы равномерного распределения случайной величины с помощью критерия хи-квадрат	Исследует последовательности на равномерность распределения в ней сгенерированных чисел.
Оценка математического ожидания каждой выборки случайных чисел	Позволяет эффективнее определить, насколько выборка соответствует равномерному распределению. Такой подход считается более надежным, поскольку он учитывает не только значения выборки, но и их отклонения от ожидаемых значений.

Продолжение таблицы 2

1	2
Проверка кумулятивных сумм	Вычисляется максимальное отклонение накопленной суммы элементов последовательности от начальной точки отсчета. Определяемый дефект – слишком много нулей или единиц в начале последовательности.
Частотный тест	Проверяет соотношение распределения нулей и единиц. Определяет слишком ли много нулей или единиц в последовательности.
Проверка случайных отклонений	Определение отклонений от ожидаемого числа посещений различных состояний при произвольном обходе.
Тест на равномерность битов	Определяет количество непрерывных серий одинаковых битов на всей длине последовательности. Определяемый дефект – колебание потока бит слишком быстрое или медленное.
Тест «блоков» в подпоследовательностях	В данном тесте определяется самая длинная серия единиц внутри блока длиной m бит. Показывает отклонение максимальных длин серий единиц от теоретического закона распределения.
Спектральный тест	Предназначен для исследования периодических свойств последовательности битов на основе высот выбросов преобразования Фурье.
Проверка аппроксимированной энтропии	Определяет меру согласования наблюдаемого значения энтропии, исследуемой СП с ожидаемым значением. Исследует неравномерность распределения m-битных пересекающихся серий.
Проверка рангов матриц	Определяется ранг матрицы, тем самым исследуется линейная независимость подстрок фиксированной длины, составляющих первоначальную последовательность.
Частотный тест в подпоследовательностях	Тест определяет количество единиц внутри блока длиной M бит. Определяет действительно ли частота повторения единиц в блоке длиной M бит приблизительно равна M÷2.
Универсальный тест Маурера	Определяется число бит между одинаковыми шаблонами в исходной последовательности. Исследуется сжимаемость последовательности.

Продолжение таблицы 2

1	2
Проверка непересекающихся шаблонов	В данном тесте подсчитывается количество заранее определённых шаблонов, найденных в исходной последовательности. Исследует частоту встречи определенных шаблонов.
Покер тест	Проверяет равномерность распределения символов в исследуемой последовательности, анализируя различные комбинации чисел в подпоследовательностях.
Тест «стопка книг»	Проверяет равномерность распределения числе в исследуемой последовательности, на основе алгоритма «перемести на передний план» преобразования данных для сжатия.
Графические тесты	
Гистограмма распределения элементов последовательности	Обеспечивает возможность оценить равномерность распределения чисел в последовательности и выявить частоту появления определенных чисел, данный тест также позволяет определить случайность последовательности по разбросу частот появления символов, который должен стремиться к нулю.
Распределение на плоскости (с фильтрующей процедурой и без)	Данный тест предназначен для определения зависимостей между элементами исследуемой последовательности. Если зависимость между элементами отсутствует, точки должны располагаться хаотично, иначе наблюдаются узоры.
Проверка серий	Основанный на анализе частоты появления нулей, единиц и серий, состоящих из k бит. Данный тест предназначен для оценки равномерности распределения символов в исследуемой последовательности.
Проверка на монотонность	Данный тест оценивает равномерность распределения символов в последовательности, используя анализ длины участков, на которых элементы последовательности не возрастают и не убывают.

Продолжение таблицы 2

1	2
Автокорреляционный тест	Основной целью данного теста является определение корреляции между сдвинутыми копиями последовательности с целью выявления возможной зависимости между подпоследовательностями, составляющими анализируемую последовательность.
Спектральный тест	Основная цель данного теста заключается в проверке равномерности распределения 0 и 1 в анализируемой последовательности. Для этого используется анализ высоты выбросов, полученных в результате преобразования Фурье.

Также при анализе систем аналогов, были выявлены необходимые требования к системе, которые позволят ей быть не только эффективной, но универсальной:

- наличие встроенных генераторов псевдослучайных последовательностей;
- наличие эмпирических и теоретических критериев;
- наличие оценочных и графических тестов различного качества;
- градация тестов по их силе;
- определение области тестирования;
- настройки параметров тестов;
- настройки параметров тестирования;
- база данных тестирования;
- возможность получения интегральной оценки;
- возможность тестирования нескольких последовательностей;
- отчет по каждому тесту.

Во второй главе приведен логический проект разработанной системы, описанный с использованием нотации UML.

Методология UML является мощным средством проектирования, устранившим недостатки более ранних методологий, в том числе и основной недостаток SADT-методологии – отсутствие объектно-ориентированного представления моделей сложных систем. Основная задача, которая стояла при создании проекта – отобразить функциональность системы.

Разработка проекта системы выполнена в бесплатной среде UML-моделирования Draw.io. На рисунке 1 приведена диаграмма вариантов использования разработанной системы.

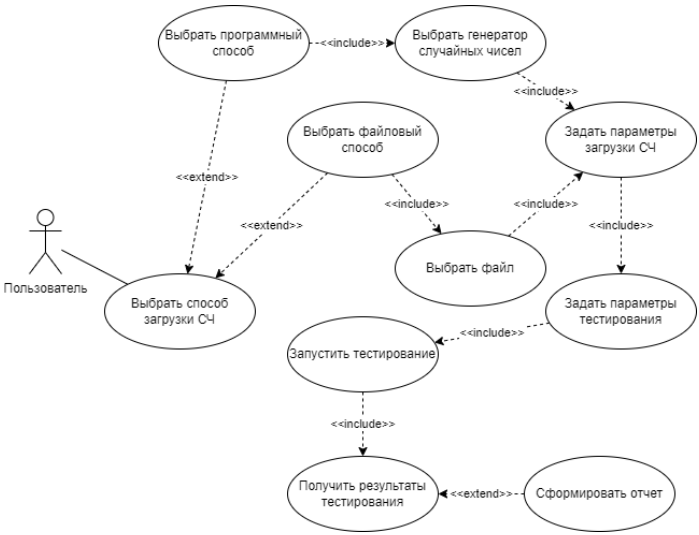

Рисунок 1 – Диаграмма вариантов использования системы

Был описан выбор комплекса программных средств, таких как: язык программирования, среда разработки, библиотеки для разработки приложения и СУБД, для реализации системы была выбрана среда IntelliJ IDEA и язык Java, а в качестве базы данных был выбран PostgreSQL.

Также приведены схемы и описание алгоритмов тестирования сгенерированных последовательностей чисел представленных в таблице 2.

Был представлен и обоснован комплекс технических средств.

В третей главе была реализована система, интерфейс которой представлен на рисунке 2.

Также исследованы три различных типа генератора при различных параметрах тестирования, основываясь на свойствах «случайности» исследуемых генераторов. В результате выдается процент пройденных тестов от их общего числа и результаты каждого теста, а также отображение графических тестов (рисунок 3-4).

В качестве контрольного примера используется демонстрация работы системы на следующих генераторах случайных чисел, которые имеют градацию по качеству генерирования случайных чисел

(приближению генерируемой последовательности чисел к истинно случайной):

- мультипликативный генератор;
- линейный конгруэнтный генератор;
- SecureRandom генератор.

Проверяемые свойства взятых генераторов:

- статистические свойства мультипликативного генератора, так как у него не очень большой период генерирования последовательности, что говорит о не очень хороших статистических свойствах данного генератора, то он не должен проходить батарею тестов, реализованную в программе испытания;
- свойства линейных конгруэнтных генераторов, у которых младшие биты генерируемых чисел, часто далеки от случайности;
- криптостойкость взятого нами SecureRandom генератора, данная тестовая последовательность должна проходить, все тесты, при любой разрядности чисел, то есть младшие биты генерируемой последовательности, тоже должны проходить испытания батареей тестов.

В результате исследования все статистические свойств выбранных генераторов подтвердились.

Представим исследования остальных генераторов, которое были реализованы в нашей системе в виде таблицы 3, используемые параметры: объем выборки = 50000, количество выборок = 100, разрядность выбирается в зависимости от типа генератора, уровень значимости тестов = 0,01, число инициализации генерируется генератором начального числа основанном на текущей дате, реализация из ГОСТ Р ИСО 24153-2012. Самыми качественными генераторами псевдослучайных чисел, которые прошли все статистические тесты, оказались генератор «Вихрь Мерсенна», KISS генератор и Java SecureRandom генератор, их качество, описанное в первой главе, подтвердилось в процессе исследований, данные генераторы могут быть использованы для различных задач, например, для моделирования методом Монте-Карло.

Таким образом, были проведена проверка эффективности системы на разработанных нами контрольных примерах, система исследовала все свойства взятых нами генераторов случайных чисел при разных параметрах системы, что подтверждает её качество и универсальность, также быстроту работы системы, было вычислено, что при параметрах объема одной выборки равной 1310720 чисел и общем количестве таких выборок равной 100, и максимальной

разрядности генерируемых чисел равной 31, система проводит все вычисления приблизительно за 2 минуты, из-за распараллеливания алгоритмов внутри тестов, это является хорошим показателем для заданных алгоритмов вычислений и по сравнению с другими аналогичными системами.

Таблица 3 – Результаты исследования генераторов

Генератор	Процент пройденных тестов %	Не пройденные тесты
1	2	3
ГОСТ Р ИСО 24153-2012	94,1	Проверка непересекающихся шаблонов
GFSR генератор с 3 параметрами (p = 1279, q = 418, w = 32)	88,2	Проверка линейной сложности, стопка книг
GFSR генератор с 5 параметрами (p = 521, q1 = 86, q2 = 197, q3 = 447, w = 32)	88,2	Проверка кумулятивных сумм, проверка случайных отклонений
Генератор Таусворта (на основе комбинации трех последовательностей параметров (31, 13, 12), (29, 2, 4) и (28, 3, 17))	82,4	Проверка гипотезы равномерного распределения случайной величины с помощью критерия хи-квадрат, проверка случайных отклонений, проверка непересекающихся шаблонов
Генератор «Вихрь Мерсенна» (с параметрами (624, 397, 31, 32, 0x9908b0df, 11, 7, 15, 18, 0x9d2c5680, 0xefc60000))	100	-
LCD v1 (a = 1664525, c = 1, m = 2^{32})	94,1	Тест серий битов
LCD v2 (a = 2100005341, c = 15, m = 2^{32})	88,2	Тест серий битов, проверка непересекающихся шаблонов
KISS генератор (на основе двух GFSR и 1 мультипликативного генератора)	100	-
Java Random генератор	94,1	Проверка непересекающихся шаблонов

Продолжение таблицы 3

1	2	3
Java SecureRandom генератор	100	-
Java SplittableRandom генератор	88,2	Стопка книг, проверка непересекающихся шаблонов
Мультипликативный генератор v1 (a = 0.011, $m = 2^{32}$)	23,5	Проверка гипотезы равномерного распределения случайной величины с помощью критерия хи-квадрат, оценка математического ожидания каждой выборки случайных чисел, проверка кумулятивных сумм, проверка на равномерность битов, частотный тест, тест серий битов, спектральный тест, проверка аппроксимированной энтропии, частотный тест в подпоследовательностях, универсальный статистический тест Маурера, проверка непересекающихся шаблонов, стопка книг, покер-тест
Мультипликативный генератор v4 (a = 214013, $m = 2^{32}$)	58,8	Проверка кумулятивных сумм, частотный тест, спектральный тест, проверка аппроксимированной энтропии, тест «блоков» в подпоследовательностях, проверка непересекающихся шаблонов, стопка книг
Мультипликативный генератор v5 (a = 16807, $m = 2^{31}$)	94,1	Тест «блоков» в подпоследовательностях

Продолжение таблицы 3

1	2	3
Мультипликативный генератор v2 (a = 11, m = 2^{32})	35,3	Проверка гипотезы равномерного распределения случайной величины с помощью критерия хи-квадрат, оценка математического ожидания каждой выборки случайных чисел, проверка кумулятивных сумм, проверка на равномерность битов, частотный тест, тест серий битов, проверка аппроксимированной энтропии, проверка линейной сложности, частотный тест в подпоследовательностях, проверка непересекающихся шаблонов, стопка книг
Мультипликативный генератор v3 (a = 37, m = 2^{32})	41,2	Проверка гипотезы равномерного распределения случайной величины с помощью критерия хи-квадрат, оценка математического ожидания каждой выборки случайных чисел, проверка кумулятивных сумм, проверка на равномерность битов, частотный тест, тест серий битов, проверка аппроксимированной энтропии, частотный тест в подпоследовательностях, проверка непересекающихся шаблонов

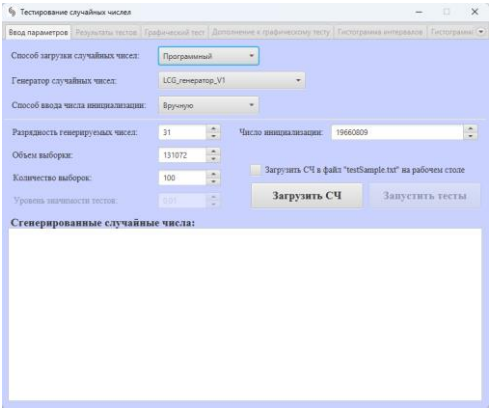

Рисунок 2 – Интерфейс разработанной системы

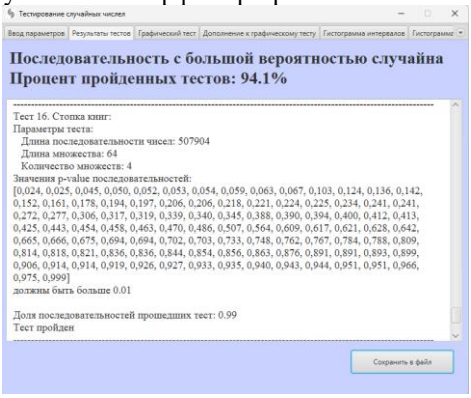

Рисунок 3 – Результаты первого тестирования

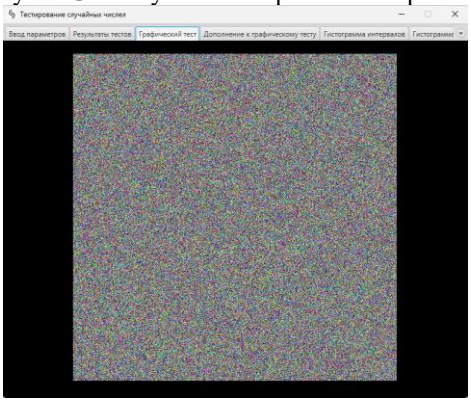

Рисунок 4 – Результат графического тестирования

В заключении сформулированы основные выводы, перечислены полученные в работе результаты.

ОСНОВНЫЕ ВЫВОДЫ И РЕЗУЛЬТАТЫ

1. Разработана эффективная и универсальная автоматизированная система, в которой реализованы статистические и графические алгоритмы тестирования, представленные в таблице 2. В системе реализована генерация различных псевдослучайных последовательностей для исследования. Выявляет отклонения генерируемых последовательностей от случайности, если таковы имеются и дает общую оценку о свойствах исследуемых генераторов. Представляется отчет о параметрах тестов и самом тестировании генераторов.

2. В результате тестирования генераторов были исследованы их свойства и определенные отклонения от случайности.

3. Произведено исследование, которое позволило выявить «лучшие» генераторы чисел из исследуемых и отсеять «худшие».