<https://eljob.ru/test/1011_7>

1. **В чем состоит особенность потоковых шифров?**

Особенность поточного шифра - иной подход к процессу шифрования в сравнении с блочным типом шифра. преобразующий каждый символ *mi* открытого текста в символ шифрованного, *ci*, зависящий от ключа и расположения символа в тексте.

Характерной особенностью поточных шифров есть *побитная обработка информации*. При этом шифрование и дешифрование может обрыватся в произвольный момент времени. И как только связь восстановлена можно продолжать процедуру без проблем. Шифрование происходит путем логических операций над битом ключа и битом исходного текста. Это происходит по тому, что сколько бы мы не создавали шифрующих битов, все равно накладываться будет один бит шифрующего на один бит исходного путем комбинации из функций XOR и отрицаний. Классический пример шифра Вернама показана на рис.1. Такие шифры отлично подходят для локальный сетей, где решают часть [проблем защиты информации в сетях](http://infoprotect.net/note/Problemyi_zasccityi_informacii_v_setyah).

Поточные шифры называют шифры гаммирования. Также само шифрование есть [методом защиты информации](http://infoprotect.net/note/Metodyi_organizacii_zasccityi_informacii). Эти шифры в разы быстрее своих конкурентов — блочных шифров, если оно реализовано аппаратно. Если же реализация программная, здесь скорость может быть даже меньше блоковых шифров. Функция которая формирует гамму, руководствуется трёма компонентами:

* Ключ.
* Номер текущего шага шифрования.
* ближние биты исходного или зашифрованного текста от текущей позиции.

1. **В чем состоят преимущества и недостатки синхронных и асинхронных потоковых шифров?**

*Синхронные потоковые шифры (СПШ)* характеризуются тем поток клю-чей генерируется независимо от открытого текста и шифртекста. Главное свойство СПШ – нераспространение ошибок. Ошибки отсутствуют, пока ра-ботают синхронно шифровальное и дешифровальное устройства отправителя и получателя информации. Один из методов борьбы с рассинхронизацией – разбить отрытый текст на отрезки, начало и конец которых выделить вставкой контрольных меток (специальных маркеров).

Синхронные потоковые шифры уязвимы к *атакам на основе изменения отдельных бит шифртекста*.

В *самосинхронизирующихся потоковых шифрах* символы ключевой гаммы зависят от исходного секретного ключа шифра и от конечного числа последних знаков зашифрованного текста. Основная идея заключается в том, что внутреннее состояние генератора потока ключей является функцией фиксированного числа предыдущих битов шифртекста. Поэтому генератор потока ключей на приемной стороне, приняв фиксированное число битов, автоматически синхронизируется с генератором гаммы.

Недостаток этих потоковых шифров – распространение ошибок, так как искажение одного бита в процессе передачи шифртекста приведет к искажению нескольких битов гаммы и, соответственно, расшифрованного сообщения.

1. **Какими свойствами должен обладать генератор псевдослучайных чисел для использования в криптографических целях?**

**(2) порождаемая последовательность должна быть "почти " неотличима от действительно случайной**

**(3) вероятности порождения различных значений ключевой последовательности должны быть равны**

**(4) для того, чтобы только законный получатель мог расшифровать сообщение, необходимо, чтобы при получении потока ключевых битов ki использовался и учитывался некоторый секретный ключ, причем вычисление числа ki+1 по известным предыдущим элементам последовательности ki без знания ключа должно быть сложной задачей**

1. **Дать характеристику линейным конгруэнтным генераторам. Области их применения.**

Часто используемый алгоритм генерирования (программно или аппа-ратно) ПСП реализуется на основе так называемого линейного конгруэнтного генератора, описываемого следующим рекуррентным соотношением:

xt+1 = (a\*xt + c) mod n, (6.1)

где: xt и xt+1 – соответственно t-й (предыдущий) и (t+1)-й (текущий, вычисляе-мый) члены числовой последовательности; а, с и n – константы. Период такого генератора (период ПСП) не превышает n.

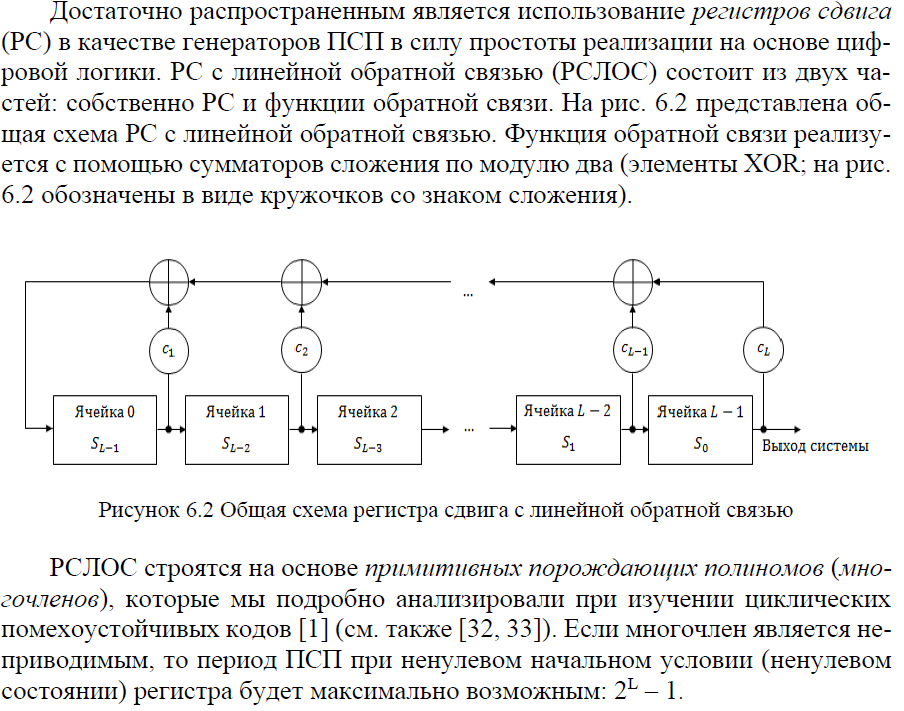
Если параметры a, b и c выбраны правильно, то генератор будет порож-дать случайные числа с максимальным периодом, равным c. При программной реализации значение с обычно устанавливается равным 2b-1 или 2b, где b – длина слова в битах. Достоинством линейных конгруэнтных генераторов псевдослучайных чи-сел является их простота и высокая скорость получения псевдослучайных зна-чений. Линейные конгруэнтные генераторы находят применение при решении задач моделирования и математической статистики, однако в криптографиче-ских целях их нельзя рекомендовать к использованию, так как специалисты по криптоанализу научились восстанавливать всю последовательность ПСЧ по нескольким ее значениям.

Генератор практически не используются в криптографии в силу низкой криптостойкости. Тем не менее, полезны для решения задач моделирования.

Комбинации нескольких (чаще двух) линейных конгруэнтных генерато-ров позволяют значительно повысить период ПСП. Б. Шнайер, например, при-водит данные о том, как на 32-разрядных ПК реализовать генератор в виде комбинации двух, каждый из которых обеспечивает период соответственно 231 – 85 и 231 – 249, а комбинированный генератор позволяет достичь периода ПСП, равного произведению указанных чисел [4].

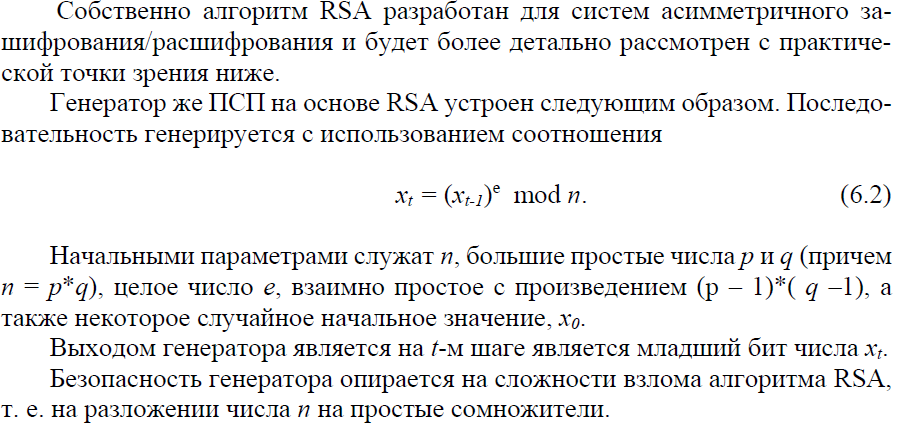
**5. Значения *x*0, *x*1, *x*2, *x3*, полученные с помощью линейного конгруэнтного генератора, равны соответственно: 1, 12, 3, 6. Найти параметры *а*, *с* и *n* генератора ПСЧ, удовлетворяющие (6.1).**

**6. Представить общую структурную схему генератора ПСП на основе регистров сдвига с линейной обратной связью. Пояснить особенности его функционирования.**

 **7. Синтезировать структурную схему генератора ПСП на основе реги-стров сдвига с линейной обратной связью, формально обозначаемого следующим образом: а) 3210, б) 420, в) 5410, г) 520, д) 84320. Составить таблицу состояний генератора и определить период ПСП.**

**8. Определить первые 12 бит ПСП, задаваемого формально в виде чисел 5410, если начальные состояния ячеек (слева-направо) соответствуют последовательности 10101.**

**9. Как устроен генератор ПСП на основе RSA? На чем основана крипто-стойкость реализуемого алгоритма?**



**10. Вычислить *х*1, х5 , *х*9, *х*11 по методу генерации псевдослучайных чисел BBS, если *p* = 11, *q* = 19, *х* = 3.**

**11. Пояснить базовый алгоритм, реализованный в шифре RC4.**

[**https://ru.wikipedia.org/wiki/RC4**](https://ru.wikipedia.org/wiki/RC4)

Алгоритм RC4, как и любой потоковый шифр, строится на основе генератора псевдослучайных битов (генератора ПСП). На вход генератора записывается ключ, а на выходе читаются псевдослучайные биты. Длина ключа может составлять от 40 до 2048 бит.

Ядро алгоритма состоит из функции генерации ключевого потока. Другая часть алгоритма – функция инициализации, которая использует *ключ перемен-ной длины Ki* для создания начального состояния генератора ключевого потока.

В основе алгоритма – размер блока или слова, определяемый параметром *n*. Обычно *n* = 8, но можно использовать и другие значения. Внутренне состо-яние шифра определяется массивом слов (*S*-блоком) размером 2n. При *n* = 8 элементы блока представляют собой перестановку чисел от 0 до 255, а сама перестановка зависит от ключа переменной длины. Другими элементами внут-реннего состояния являются 2 счетчика (каждый размером в одно слово; обо-значим их i и *j*) с нулевыми начальными значениями. В основе вычислений лежит операция по *mod* 2n.

Генератор ключевого потока RC4 переставляет значения, хранящиеся в *S*, и каждый раз выбирает различное значение из *S* в качестве результата. В одном цикле RC4 определяется одно *n*-битное слово *K* из ключевого потока, которое в последующем суммируется с исходным текстом для получения зашифрован-ного текста. Эта часть алгоритма называется генератором ПСП. При n = 8 для генерации случайного байта выполняются операции, представленные листин-гом 6.1.

i = (i + 1) mod 256;

j = (j + Si) mod 256;

поменять местами Si и Sj;

a = (Si + Sj) mod 256;

K = Sa

Листинг 6.1 Псевдокод для генерации байта ПСП

Байт *K* используется в операции XOR с открытым текстом для получения 8-битного шифртекста или для его расшифрования.

Так же достаточно проста и инициализация *S*-блока. Этот алгоритм ис-пользует ключ, который подается на вход пользователем. Сначала *S*-блок за-полняется линейно: *S*0 = 0, *S*1 = 1, …, *S*255 = 255. Затем заполняется секретным ключом другой 256-байтный массив. Если необходимо, ключ повторяется многократно, чтобы заполнить весь массив: *K*0, *K*1, …, *K*255. Далее массив *S* перемешивается путем перестановок, определяемых ключом. Действия выпол-няются в соответствии с псевдокодом, представленным листингом 6.2.

**12. Пояснить принципы формирования истинных случайных последова-тельностей, основанных на «естественных случайностях».**

Наилучшие характеристики будут иметь генераторы случайных чисел, ос-нованные на «естественных случайностях», свойственных, например, процес-сам в радиоэлектронной аппаратуре, в системах телекоммуникаций, в приемах работы с клавиатурой операторов и др.