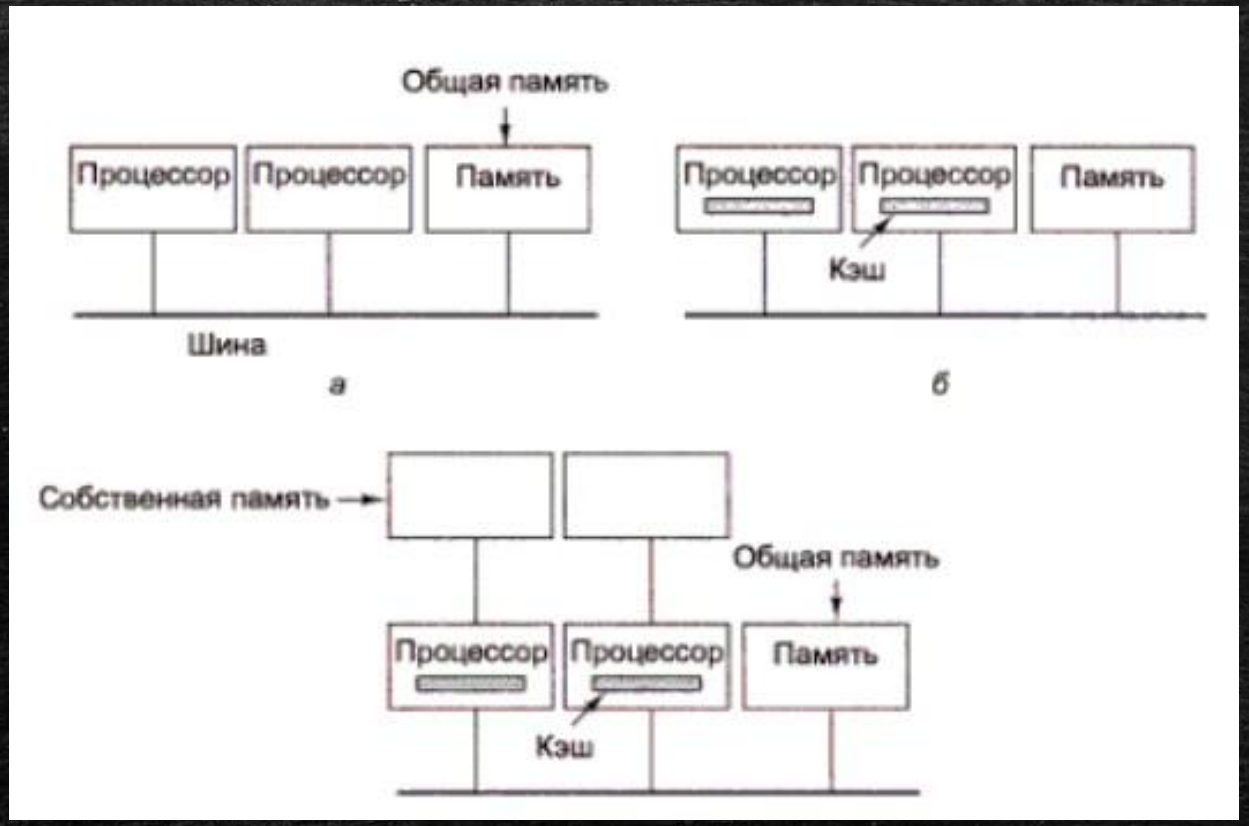
Лекция 7. вроде бы

Вроде бы и архитектура. Но математики много.

Мультипроцессоры UMA



Cache coherency protocol

- Write through
- Промах чтения (загрузка строки)
- Попадание чтения (-)
- Промах записи (запись в память)
- Попадание записи(обновление кэша, зпись в память)

MESI

- Invalid
- Shared
- Exclusive
- Modified

Нужна коммутация

- Перекрестная коммутация с n процессорами и k блоками памяти

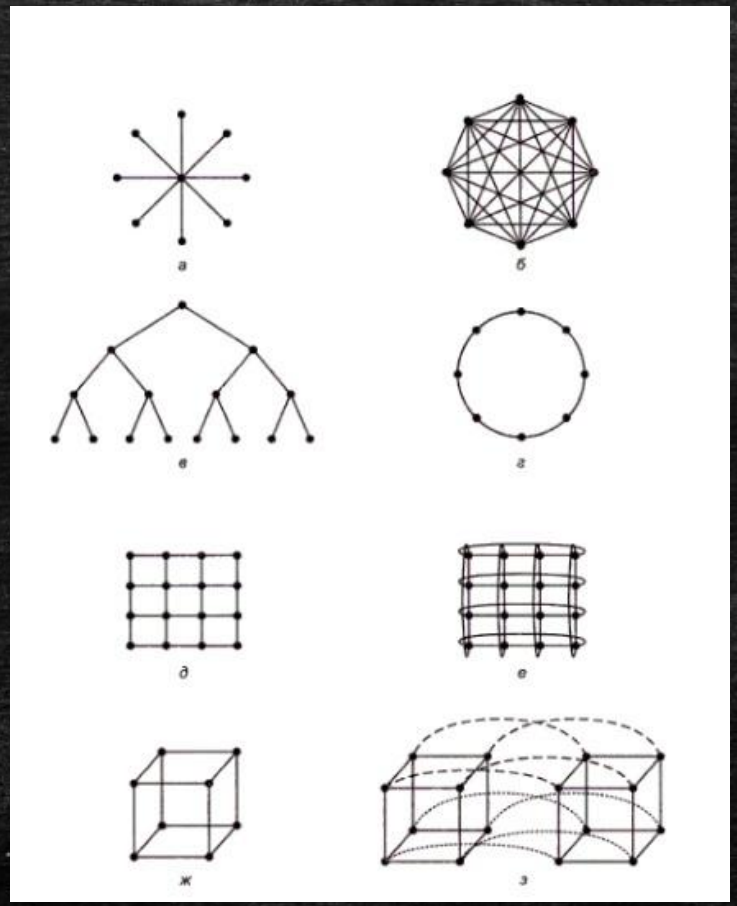
Мультикомпьютеры

- Плохая масштабируемость
- Конкуренция за доступ к памяти
- Send & recieve

Коммуникационные сети

- Топология КС –схема рзмещений линий связи и коммутаторов. Часто – неорграф (V,E), V – линии связи, E – коммутаторы.
- У каждого узла степень, влияющая на отказоустойчивость
- Диаметр – большую задержку при передаче пакетов
- Пропускная способность- объем данных в секунду
- Размерность – количество вариантов перехода.

Многомерность



MPP

- Обычные процессоры с дорогим ПО
- Огромные объемы IO
- Отказоустойчивость

Кластерные компьютеры

- Несколько ПК или р\с
- Централизованный и децентрализованные

Message Parsing Interface

- Пользователь создает процессы.
- Коммуникаторы
- Тип данных
- Операции отправки и получения
- Виртуальные топологии

- MPI_Send(буфер, число_элементов, тип_данных, получатель, тег, коммуникатор)
- MPI_Recv(&буфер, число_элементов, тип_данных, отправитель, тег, коммуникатор, &статус)

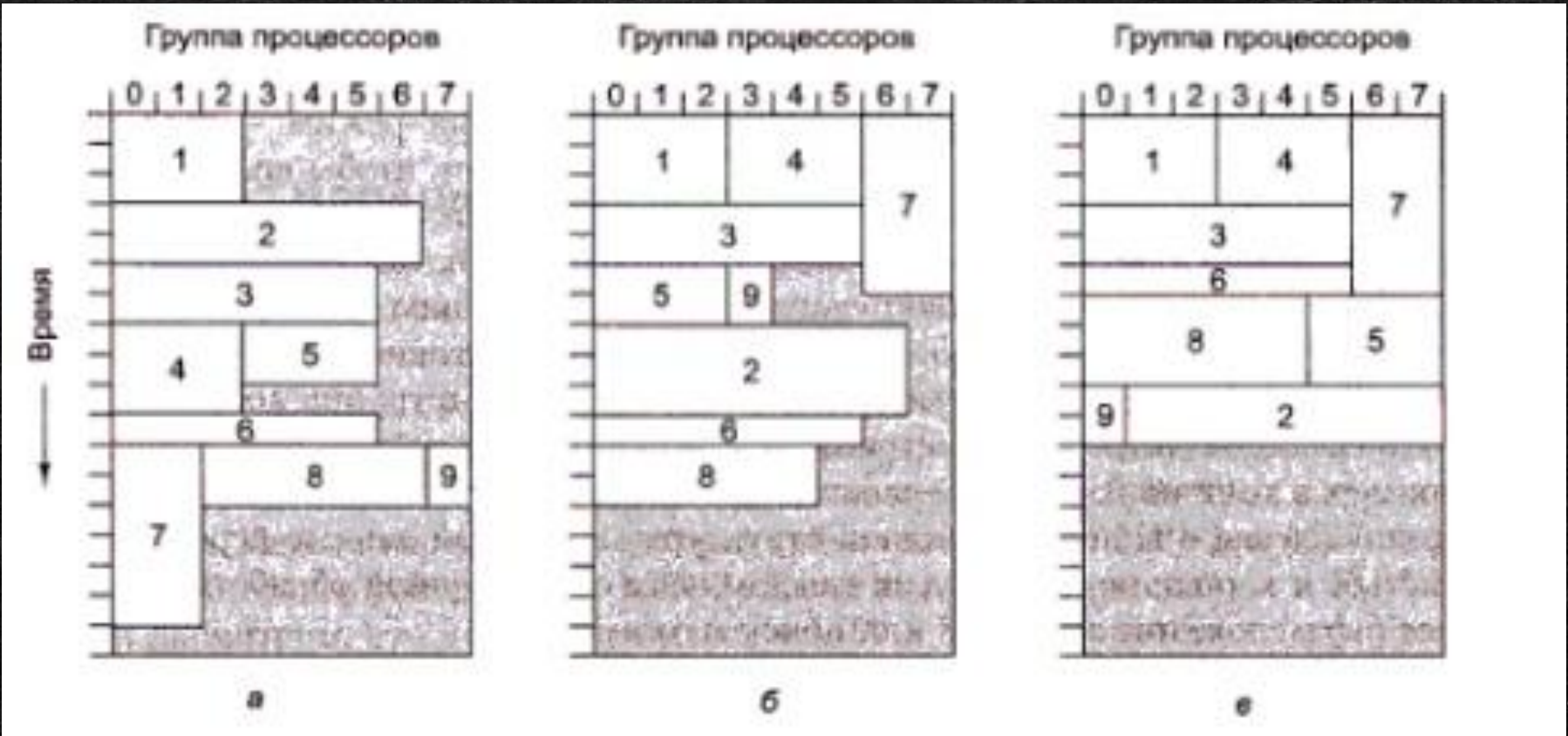
Коммуникационные режимы

- Синхронный
- Буферизация
- Стандарт
- Готовность

MPI-2

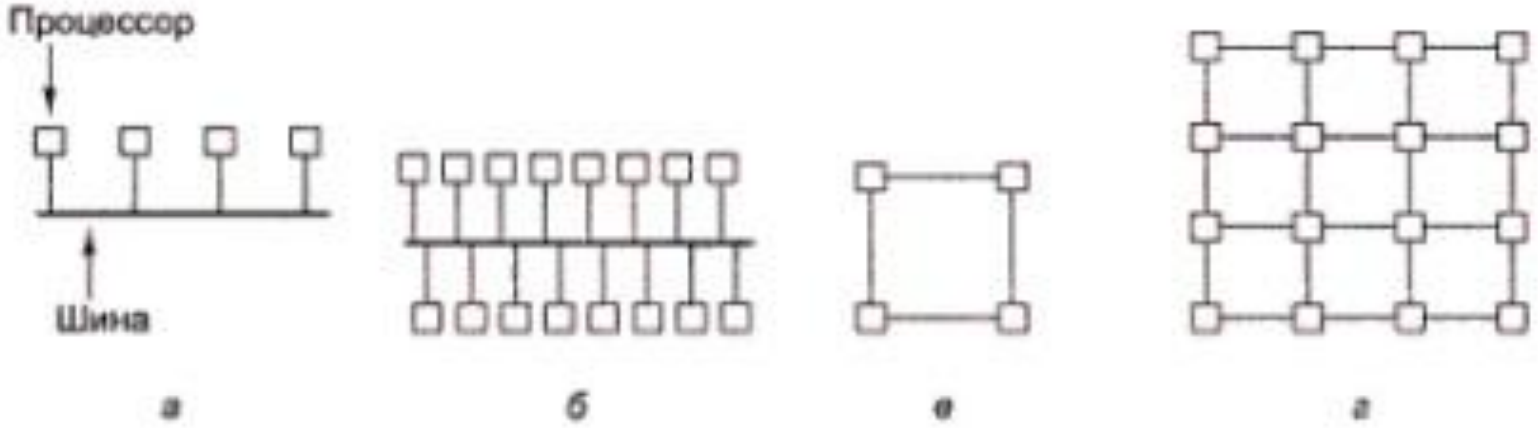
- ДП
- Удаленный доступ
- Масштабируемый IO

Планирование



Производительность

- Добавить процессоры, но умно. Нужно соблюдать масштабируемость.
- Решетка – хорошо.


Процессор
Шина
а
б
в
г

- Пропускная способность растет, время запаздывания – тоже.
- В идеале все должно быть неизменно
- По факту – как есть.

Как сократить или замаскировать время запаздывания?

- Репликация
- Упреждающая выборка
- Многопоточность
- Неблокирующие записи

Disturbed computing

- Очень большой, интернациональный слабо связанный гетерогенный кластер
- Цель – создать инфраструктуру, которая бы из нескольких организаций сделала бы единую виртуальную организацию.
- Многомерная система с одноранговыми узлами.
- Куча ресурсов и организаций

Моделирование Системы распределенных вычислений

- Уровень инфраструктуры – физ ресурсы, из которых построена система

Моделирование Системы распределенных вычислений

- Уровень инфраструктуры – физ. ресурсы, из которых построена система
- Уровень ресурсов – управление отдельными ресурсами

Моделирование Системы распределенных вычислений

- Уровень инфраструктуры – физ. ресурсы, из которых построена система
- Уровень ресурсов – управление отдельными ресурсами
- Уровень коллективов – исследование, посредничество, управление ресурсами.

Моделирование Системы распределенных вычислений

- Уровень инфраструктуры – физ. ресурсы, из которых построена система
- Уровень ресурсов – управление отдельными ресурсами
- Уровень коллективов – исследование, посредничество, управление ресурсами.
- Уровень приложений – приложения, которые совместно используют ресурсы

Безопасность

- Однократная регистрация в системе
- Нужны стандарты
- Global Grid Forum, OGSA

Категрории стандартизированных служб

- Службы инфраструктуры (обеспечивают взаимодействие между ресурсами).
- Службы управления ресурсами (резервирование и освобождение ресурсов).
- Службы данных (копирование и перемещение данных туда, где они нужны).
- Контекстные службы (описание требуемых ресурсов и политик их использования).
- Информационные службы (получение информации о доступности ресурса).
- Службы самоконтроля (поддержание заявленного качества услуги).
- Службы защиты (применение политик безопасности).
- Службы управления выполнением (управление потоком задач)

Математика

- Задача
- д - доля последовательных расчетов, 1-д - можно идеально распараллелить на p задействованных узлах.
- Тогда справедлив закон Амдала – ускорение, которое получаем на системе из P процессоров по сравнению с однопроцессорной машине не превышает $S_p = \frac{1}{д + (1-д)p}$

$\alpha \backslash p$	10	100	1000
0	10	100	1000
10%	5.263	9.174	9.910
25%	3.077	3.883	3.988
40%	2.174	2.463	2.496

Следствия закона Амдала

- $S_p = \frac{1}{\partial + (1-\partial)p}$
- При доле последовательных вычислений K прирост производительности не превысит 1/k
- Прирост эффективности ограничен сверху для ∂ !=0
- Если учесть время для передачи данных, то зависимость времени вычислений от числа узлов будет иметь максимум

Математика. Квантовый компьютер

- Квантовый компьютер — гипотетическое вычислительное устройство, которое путем выполнения квантовых алгоритмов существенно использует при работе квантовомеханические эффекты, такие как квантовый параллелизм и квантовая запутанность.
- Данные в процессе вычислений представляют собой квантовую информацию, которая по окончании процесса преобразуется в классическую путём измерения конечного состояния квантового регистра. Выигрыш в квантовых алгоритмах достигается за счет того, что при применении одной квантовой операции большое число коэффициентов суперпозиции квантовых состояний, которые в виртуальной форме содержат классическую информацию, преобразуется одновременно

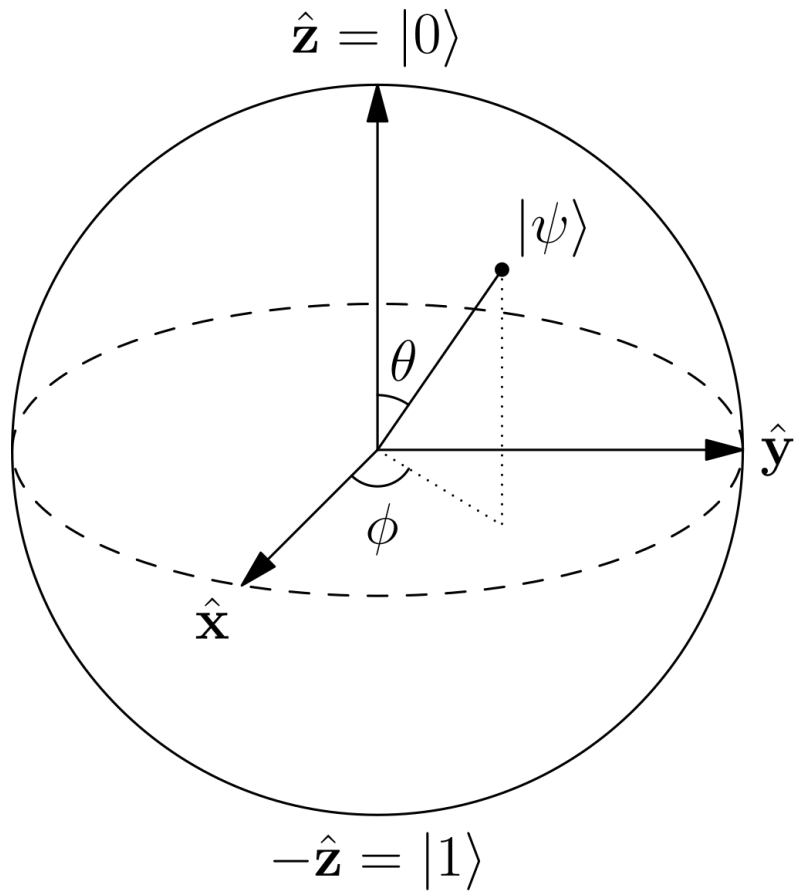
Квантовый компьютер

- Квантовая запутанность иногда называют квантовой суперпозицией.
- Цифровое устройство с аналоговой природой

- Бит – н/м единица измерения информации
- Может быть только в двух базовых состояниях - 0 или 1 и находиться только в одном из этих состояний

- Бит – н/м единица измерения информации
- Может быть только в двух базовых состояниях - 0 или 1 и находиться только в одном из этих состояний
- Квантовый бит – тоже может быть в двух основных состояниях…

- Бит – н/м единица измерения информации
- Может быть только в двух базовых состояниях - 0 или 1 и находиться только в одном из этих состояний
- Квантовый бит – тоже может быть в двух основных состояниях…
- И может находиться в комбинациях этих состояний, и может находиться во всех этих состояниях одновременно.
- $|x> =$ a$|0> + b|1>, a^2 + b^2 = 1$ и a, b – комплексные числа


$\hat{\mathbf{z}} = |0\rangle$
$|\psi\rangle$
θ
$\hat{\mathbf{y}}$
ϕ
$\hat{\mathbf{x}}$
$-\hat{\mathbf{z}} = |1\rangle$

- При измерениях кубиты случайно переходят в одно из двух собственных состояний(с вероятностью A^2 и B^2)
- Так же кубиты могут быть как-то связаны между собой так, что изменение одного кубита, может повлечь изменение другого.
- Ну например если у меня в системе L кубитов, то имеется 2^L независимых состояний. Стало быть L – кубит – 2^L классических состояний.
- Фотоны, излированные атомы и ионы

Пример

- $x = 0.8|0> - 0.6|1>$
- P(0) = 0.64
- P(1) = 0.36
- В результате измерения получим новое квантовое состояние |0> и при следующих вычисления P = 1

Запутывание

- $x = \frac{1}{\sqrt{2}} 00 + \frac{1}{\sqrt{2}} |11>$
- Измеряя первый кубит по правилам получим , что наша система спроецируется либо на 00, либо на 11.
- .И мы знаем второй кубит, не измеряя его.

Почему гипотетическое?

- необходимо обеспечить высокую точность измерений;
- внешние воздействия могут разрушить квантовую систему или внести в неё искажения

RSA

- Назовем функцию F односторонней, если:

1. При известном x мы сможем посчитать F(x)
2. При известном y=F(x) мы не сможем посчитать эффективно F(x)

- В основе лежит задача факторизации произведения двух больших простых чисел
- Шифрование – возведение в степень по модулю большого числа x
- Дешифрование – подсчет ϕ(x) – функции Эйлера

Функция Эйлера

- $\phi(x)$ = количество чисел от 1 до x-1, взаимно простых с x
- $\phi(6) = 2$, $\phi(25) = 20$, $\phi(11) = 10$

- Участник Боб и Алиса располагает как открытым ключом так и закрытым ключом
- Ключ – пара целых чисел
- Каждый создает ключ сам.
- Закрытый хранится при себе, открытый – в открытом доступе
- Для каждого участника открытый и закрытый ключ образуют взаимно-обратные функции.
-

Выбираются два различных случайных простых числа p и q заданного размера (например, 1024 бита каждое).

Вычисляется их произведение n=p*q, которое называется модулем.

Вычисляется значение функции Эйлера от числа n: ϕ(n) = (p-1)(q-1)

Выбирается целое число e : 1<e< ϕ (n).Число e называется открытой экспонентой (англ. public exponent)

Слишком малые значения e, например 3, потенциально могут ослабить безопасность схемы RSA.

RSA

Вычисляется число d, мультипликативно обратное к числу e по модулю то есть число, которое бы делилось в произведении на e на ϕ(n) то есть d*e делится на ϕ(n)

Число d называется секретной экспонентой. Обычно, оно вычисляется при помощи расширенного алгоритма Евклида.

Пара (e,n) публикуется в качестве открытого ключа RSA (англ. RSA public key).

Пара (d,n) играет роль закрытого ключа RSA (англ. RSA private key) и держится в секрете.

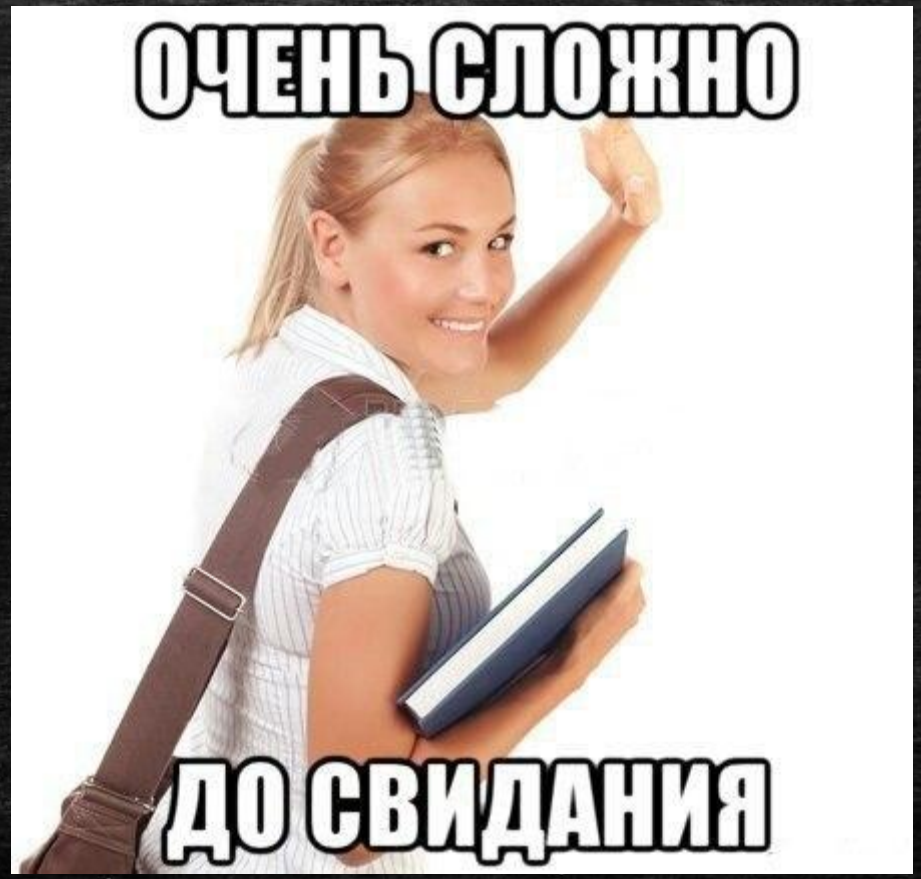
Шлем m

- Берем (e,n) у Алисы
- Берем m, возводим в e и находим остаток при делении на n\
- Отправляем ответ C
- Как приняли – берем свой закрытый ключ (d,n)
- Возводим ответ C в Степень d и ищем остаток при делении на n
- Вуаля.

Пример

Этап	Описание операции	Результат операции
Генерация ключей	Выбрать два простых различных числа	$p = 3557,$ $q = 2579$
	Вычислить произведение	$n = p \cdot q = 3557 \cdot 2579 = 9173503$
	Вычислить функцию Эйлера	$\varphi(n) = (p-1)(q-1) = 9167368$
	Выбрать открытую экспоненту	$e = 3$
	Вычислить секретную экспоненту	$d = e^{-1} \mod \varphi(n)$ $d = 6111579$
	Опубликовать *открытый ключ*	$\{e, n\} = \{3, 9173503\}$
	Сохранить *закрытый ключ*	$\{d, n\} = \{6111579, 9173503\}$
Шифрование	Выбрать текст для зашифрования	$m = 111111$
	Вычислить шифротекст	$c = E(m)$ $= m^e \mod n$ $= 111111^3 \mod 9173503$ $= 4051753$
Расшифрование	Вычислить исходное сообщение	$m = D(c) =$ $= c^d \mod n$ $= 4051753^{6111579} \mod 9173503$ $= 111111$

Коротко о квантовых алгоритмах



- Матрица Адамара
- $\frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

Алгоритм ШОРА

- Алгоритм Шора — это квантовый алгоритм факторизации (разложения числа на простые множители), позволяющий разложить число N за время O((log N)^3), затратив O(log N) кубитов
- Вероятностный алгоритм
- В 1994г Разработан Питером Шором, в 2001г IBM разложила 15 на 3*5

Принцип алгоритма Шора

- Пусть M- число , которое хотим взять и разложить на множители

1. поиск периода f(x) = p^x (mod M), a = random (<M)
2. Допустим M = 21, а в качестве p выберем 2

x		0	1	2	3	4	5	6	7
a^x	**2^x**	1	2	4	8	16	32	64	128
a^x mod M	**2^x mod 21**	1	2	4	8	16	11	1	2

EAX.ME

Видно, что r = 6. Найденный период обязан быть четным

А дальше просто Ищем НОД$\left(2^{\frac{6}{2}} \pm 1,21\right) = 3$

Пара 7,3 искомая.

- Выбрать случайное число a, меньшее M: a < M.
- Вычислить НОД(a, M). Это может быть сделано при помощи алгоритма Евклида.
- Если НОД(a, M) не равен 1, то существует нетривиальный делитель числа M, так что алгоритм завершается (вырожденный случай).
- В противном случае необходимо использовать квантовую подпрограмму поиска периода функции f(x) = a^x mod M.
- Если найденный период r является нечётным, то вернуться на шаг 1 и выбрать другое число a.
- Если $a^{(r/2)} \equiv M - 1 \pmod{M}$, то вернуться на шаг 1 и выбрать другое число a.
- Наконец, определить два значения НОД(a^(r/2) ± 1, M), которые и являются нетривиальными делителями числа M.

Недавний факт про алгоритм Шора

- Число 15 разложили с помощью 5 кубитов с вероятностью получения правильного ответа в 50%. И говорят, что

Некоторые другие алгоритмы

- Алгоритм Донча-Йожи
- Пусть f(a, b…k)- переключательная функция, которая может быть либо константой, либо сбалансированной функцией
- Нужно определить F – сбалансирована или константа
- Нужно 2^(n-1) +1 классических вычислений.

И еще

- Алгоритм Гровера решения уравнения f(x) =1
- Алгоритм Залки-Визнера моделирования эволюции
- Алгоритм Саймона решения проблемы черного ящика