Министерство образования и науки Российской Федерации

Санкт-Петербургский Государственный Политехнический Университет

—

Высшая школа программной инженерии

Направление «Фундаментальная информатика и информационные технологии»

Доклад

по дисциплине «Дискретная математика»

на тему «Криптография. Методы шифрования»

Выполнила:

студент гр. 3530202/90002 Потапова А. М.

Преподаватель: Тышкевич А. И.

Санкт-Петербург

2021

**Оглавление**

[Введение 3](#_Toc70456526)

[Основное понятие криптографии 4](#_Toc70456527)

[Составные части криптографии 5](#_Toc70456528)

[Методы шифрования 6](#_Toc70456529)

[*Симметричное шифрование* 7](#_Toc70456530)

[*Асимметричное шифрование* 8](#_Toc70456531)

[RSA 8](#_Toc70456532)

[Заключение 11](#_Toc70456533)

# **Введение**

С каждым годом компьютерная информация играет все более важную роль в нашей жизни, и все большую актуальность приобретают проблемы ее защиты.

Защита от каждого типа опасности предполагает собственные решения. Впрочем, есть и универсальные подходы, способные обезопасить данные от разных угроз. Одним из них является криптография, то есть искусство и наука создания криптосистемы, способной обеспечить информационную безопасность.

Криптография имеет дело с фактической защитой цифровых данных. Это относится к разработке механизмов, основанных на математических алгоритмах, которые предоставляют фундаментальные услуги информационной безопасности. Вы можете думать о криптографии как о создании большого инструментария, содержащего различные методы в приложениях безопасности.

# **Основное понятие криптографии**

Основным компонентом криптографии является шифрование. Сообщения шифруются и расшифровываются с помощью сложных алгоритмов, созданных комбинацией информатики и математики.

Шифрование использует алгоритм и ключ для преобразования входных данных в зашифрованные выходные данные. Этот метод защиты позволяет просматривать сообщения исключительно отправителю и получателю, поскольку зашифрованную информацию может прочесть только тот, кто имеет секретный ключ для преобразования сообщения в простой текст.

# **Составные части криптографии**

Прежде чем перейти к современным криптографическим алгоритмам, предлагаю подробнее ознакомиться с составными частями всех криптографических систем:

* **Исходное сообщение** — то, что мы хотим защитить от несанкционированного чтения/использования.
* **Зашифрованное сообщение** — это сообщение, измененное с целью скрыть его исходный смысл и сделать его "нечитаемым".
* **Криптографический алгоритм** — некий математический алгоритм, используемый для шифрования или дешифрования исходного сообщения.
* **Ключ** — вспомогательная информация, используемая криптографическим алгоритмом. Она секретна, и априори считается, что, только обладая ею, можно восстановить исходное сообщение.

# **Методы шифрования**

* *Симметричное шифрование* использует один и тот же ключ и для зашифровывания, и для расшифровывания.
* *Асимметричное шифрование* использует два разных ключа: один для зашифровывания (открытый), другой для расшифровывания (закрытый).

Эти методы решают определённые задачи и обладают как достоинствами, так и недостатками. Конкретный выбор применяемого метода зависит от целей, с которыми информация подвергается шифрованию. Рассмотрим эти методы подробнее.

## ***Симметричное шифрование***

Является самым простым алгоритмом. Криптографы часто называют его секретным ключом криптографии или общим, поскольку шифрование и расшифровка информации происходит с использованием одного и того же ключа. Симметричное шифрование подразумевает, что секретный цифровой ключ должен быть известен как получателю, так и отправителю.

Приведу простую схему реализации: есть два собеседника − А и Б, они хотят обмениваться конфиденциальной информацией. Для удобства разобью этот процесс на этапы:

1. **Генерация ключа**.

А выбирает ключ шифрования *key* и алгоритмы *E* (ф-я шифрования), *D* (ф-я расшифрования), затем посылает эту информацию Б.

1. **Шифрование и передача сообщения**.

Б шифрует сообщение m с использованием полученного ключа *key*.

*E(m, key) = encoded\_m*

И передает А полученный шифротекст *encoded\_m.*

1. **Расшифровывание сообщения**.

А, с помощью того же ключа *key*, расшифровывает шифротекст *encoded\_m.*

*E(encoded\_m, key) = m*

Таким образом если кто-то узнает наш пароль, безопасность криптосистемы тут же нарушится. Именно поэтому, используя подходы симметричного шифрования, мы должны особое внимание уделять **вопросам создания и сохранения конфиденциальности пароля.** Но, несмотря на свои ограничения и угрозу безопасности, подход до сих пор широко распространён в криптографии, так как он очень прост в работе и понимании.

## ***Асимметричное шифрование***

Широко используется во Всемирной сети. Его также называют открытым ключом криптографии. Этот алгоритм использует два ключа: открытый и закрытый.

* Открытый может быть известен многим. Расшифровать данные с его помощью невозможно. Хорошим примером открытого ключа является адрес электронной почты.
* Закрытый является секретным, используется для расшифровки сообщения, никогда не раскрывается другой стороне. Например, пароль учетной записи электронной почты является ключом к открытию электронных писем.
* Не имеет значения, какой ключ применяется в первую очередь, но для работы необходимы оба.

Для полного понимания сути асимметричного шифрования предлагаю рассмотреть алгоритм RSA — самый популярный криптографический алгоритм с открытым ключом, основывающийся на вычислительной сложности задачи факторизации больших целых чисел.

### **RSA**

Итак. Допустим, я хочу получить от вас некие данные. Мы с вам не хотим, чтобы эти данные узнал кто-то, кроме нас. И у нас нет никакой уверенности в надёжности канала передачи данных. Приступим.

1. ***Подготовка ключей.***

Я должна проделать предварительные действия: сгенерировать открытый и закрытый ключ.

* Выбираю два простых числа. Пусть это будет p = 3 и q = 7.
* Вычисляю *модуль* — произведение наших p и q: n = p × g = 21.
* Вычисляю *функцию Эйлера*: φ = (p - 1) × (q - 1) = 12.
* Выбираю число e, отвечающее следующим критериям:
  1. оно должно быть простое,
  2. оно должно быть меньше φ, остаются варианты: 3, 5, 7, 11,
  3. оно должно быть взаимно простое с φ — остаются варианты 5, 7, 11.

Выбираю e = 5. Это, так называемая, *открытая экспонента*.

Теперь пара чисел {e, n} — это мой открытый ключ. Я отправляю его вам, чтобы вы зашифровали своё сообщение. Но для меня это ещё не всё. Я должна получить закрытый ключ.

Мне нужно вычислить число d, обратное е по модулю φ. То есть остаток от деления по модулю φ произведения d × e должен быть равен 1. Запишем это в обозначениях, принятых во многих языках программирования: (d × е) % φ = 1. Или (d × 5) % 12 = 1. d может быть равно 5, но чтобы оно не путалось с e в дальнейшем повествовании, давайте возьмём его равным 17.

Пара {d, n} — это секретный ключ, его я оставляю у себя. Его нельзя сообщать никому. Только обладатель секретного ключа может расшифровать то, что было зашифровано открытым ключом.

1. ***Шифрование***

Теперь пришла ваша очередь шифровать ваше сообщение. Допустим, ваше сообщение — это число 19. Обозначим его P = 19. Кроме него у вас уже есть мой открытый ключ: {e, n} = {5, 21}. Шифрование выполняется по следующему алгоритму:

* Возводите ваше сообщение в степень e по модулю n. То есть, вычисляете 19 в степени 5
* Берёте остаток от деления на 21. Получается 10 — это ваши закодированные данные.

Полученные данные E=10, вы отправляете мне. Здесь надо заметить, что сообщение P = 19 не должно быть больше n = 21. Иначе ничего не получится.

1. ***Расшифровка***

Я получила ваши данные (E=10), и у меня имеется закрытый ключ {d, n} = {17, 21}. Обратите внимание на то, что открытый ключ не может расшифровать сообщение. А закрытый ключ я никому не говорила. В этом вся прелесть асимметричного шифрования. Начинаем раскодировать:

* Я делаю операцию, очень похожую на вашу, но вместо e использую d. Возвожу E в степень d: получаю 10 в степень 17, а далее вычисляю остаток от деления на 21 и получаю 19 — ваше сообщение.

Заметьте, никто, кроме меня (даже вы) не может расшифровать ваше сообщение (E=10), так как ни у кого нет закрытого ключа.

Стоит отметить, что не очевидно, как вычислить 10^17 % 21 «в голове», но дискретная математика обеспечивает быстрые алгоритмы модульного возведения в степень. Используя теорему Ферма, теорию конгруэнции и расширенный алгоритм Евклида, мы можем эффективно вычислить показатель магического декодирования d, который будет занимать 19, и вернуть его обратно к 10.

Таким образом, единственное, что препятствует расшифровке публично кодированного сообщения RSA, заключается в том, что они не знают (p–1) (q–1). И единственный способ узнать это число – получить его от произведения pq, что потребует факторинга, и никто не знает, как это сделать эффективно. Поэтому единственными людьми, которые могут расшифровать публично закодированные сообщения RSA, являются люди, которые создали pq, в первую очередь, потому что только они знают p и q. Существует множество примеров, но этот один из ярчайших, который объясняет, как дискретная математика неразрывно связана с криптографией.

# **Заключение**

В заключение хотелось бы отметить, что криптография порождает новые трудные математические задачи, при этом математика является каркасом, основой криптографии. По мере развития общества и его научных достижений, разрабатываются новые математические методы, приводящие к разрешению задач, которые ранее считались неразрешимыми.