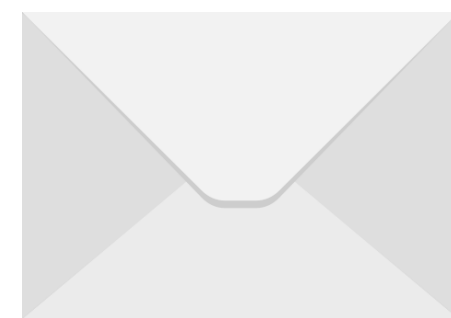
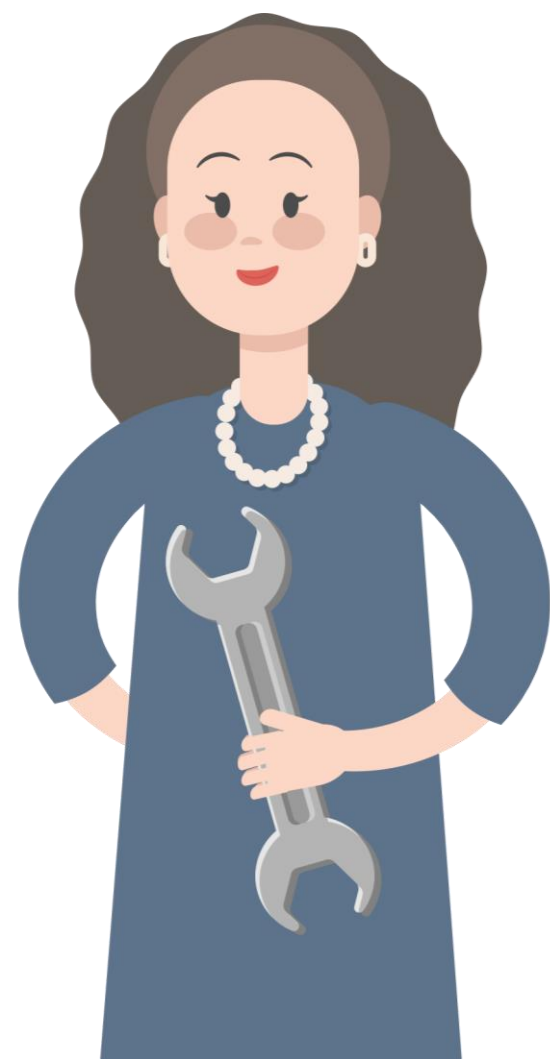




Гомоморфное шифрование

Гомоморфное шифрование





Гомоморфное шифрование

Определение: Гомоморфное шифрование — форма шифрования, позволяющая производить определённые математические действия с зашифрованным текстом и получать зашифрованный результат, который соответствует результату операций, выполненных с открытым текстом.

Гомоморфизм

Гомоморфизм (от др.-греч. - равный, одинаковый и - вид, форма) — это морфизм в категории алгебраических систем, то есть отображение алгебраической системы A , сохраняющее основные операции и основные отношения.

Отображение $f: G_1 \rightarrow G_2$ называется гомоморфизмом групп $(G_1, *) (G_2, \times)$, если оно одну групповую операцию переводит в другую:

$$f(a * b) = f(a) \times f(b).$$

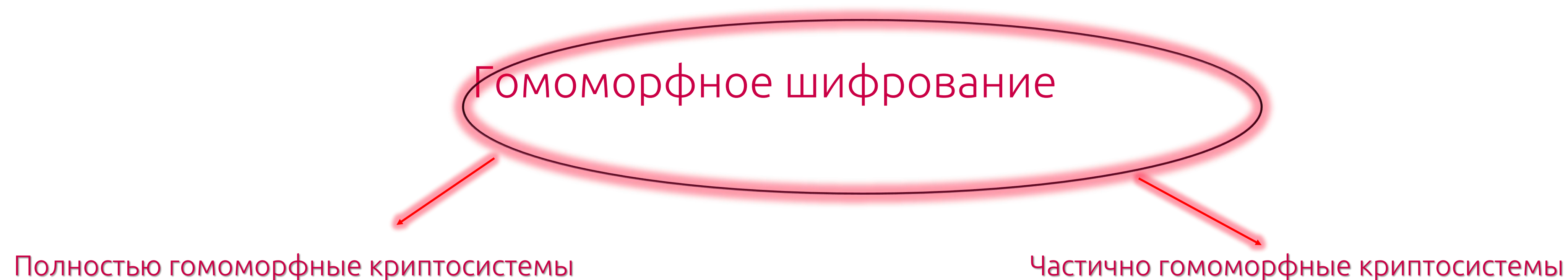
Гомоморфизм



Понятие гомоморфизма как соотношение между парой алгебраических систем начало использоваться в работах немецкого математика Фробениуса, а обобщённое определение было сформулировано Эмми Нётер в 1929 году.



Гомоморфное шифрование



- *Частично гомоморфные криптосистемы* позволяют производить только одну из операций — либо сложение, либо умножение.
- *Полностью гомоморфная криптосистема* поддерживает выполнение обеих операций, то есть, в ней выполняются свойства гомоморфизма как относительно умножения, так и относительно сложения.

История

1. Понятие *«гомоморфное шифрование»* сформировано в 1978 году Рональдом Ривестом, Леонардом Адлеманом и Майклом Дертузосом в «On Data Banks And Privacy Homomorphisms»



2. В 1982 году Шафи Гольдвассер и Сильвио Микали предложили систему шифрования, гомоморфную относительно умножения и способную зашифровать всего лишь один бит.



3. В 1999 году Паскалем Пэе предложил криптосистему, гомоморфную относительно умножения.



История

Криптосистема Джентри — первая возможная конструкция для полностью гомоморфной криптосистемы, основанная на криптографии на решетках. Была предложена Крейгом Джентри в 2009 году и поддерживает операции сложения и умножения над шифротекстом.



Полностью гомоморфное шифрование (Fully Homomorphic Encryption)

Определение: Полностью гомоморфное шифрование — криптографический примитив, представляющий собой функцию шифрования, удовлетворяющую дополнительному требованию гомоморфности относительно каких-либо операций над открытыми текстами.

Гомоморфное шифрование

- k — ключ;
- m — открытый текст;
- $Enc(k, m)$ — шифрующая функция
- $Dec(k, m)$ — дешифрующая функция

Определение: Функция Enc называется гомоморфной относительно операции сложения или умножения ($*$) над открытыми текстами m_1, m_2 , если существует алгоритм H : получив на входе пару $Enc(k, m_1)$ и $Enc(k, m_2)$ даст

$$C = H(Enc(k, m_1), Enc(k, m_2)),$$

который при дешифровании даст открытый текст $m_1 * m_2$.

Гомоморфное шифрование

Определение: Криптосистема гомоморфна относительно операции сложения, если $Dec(Enc(k, m_1) + Enc(k, m_2)) = m_1 + m_2$.

Определение: Криптосистема гомоморфна относительно операции умножения, если $Dec(Enc(k, m_1) \cdot Enc(k, m_2)) = m_1 \cdot m_2$.

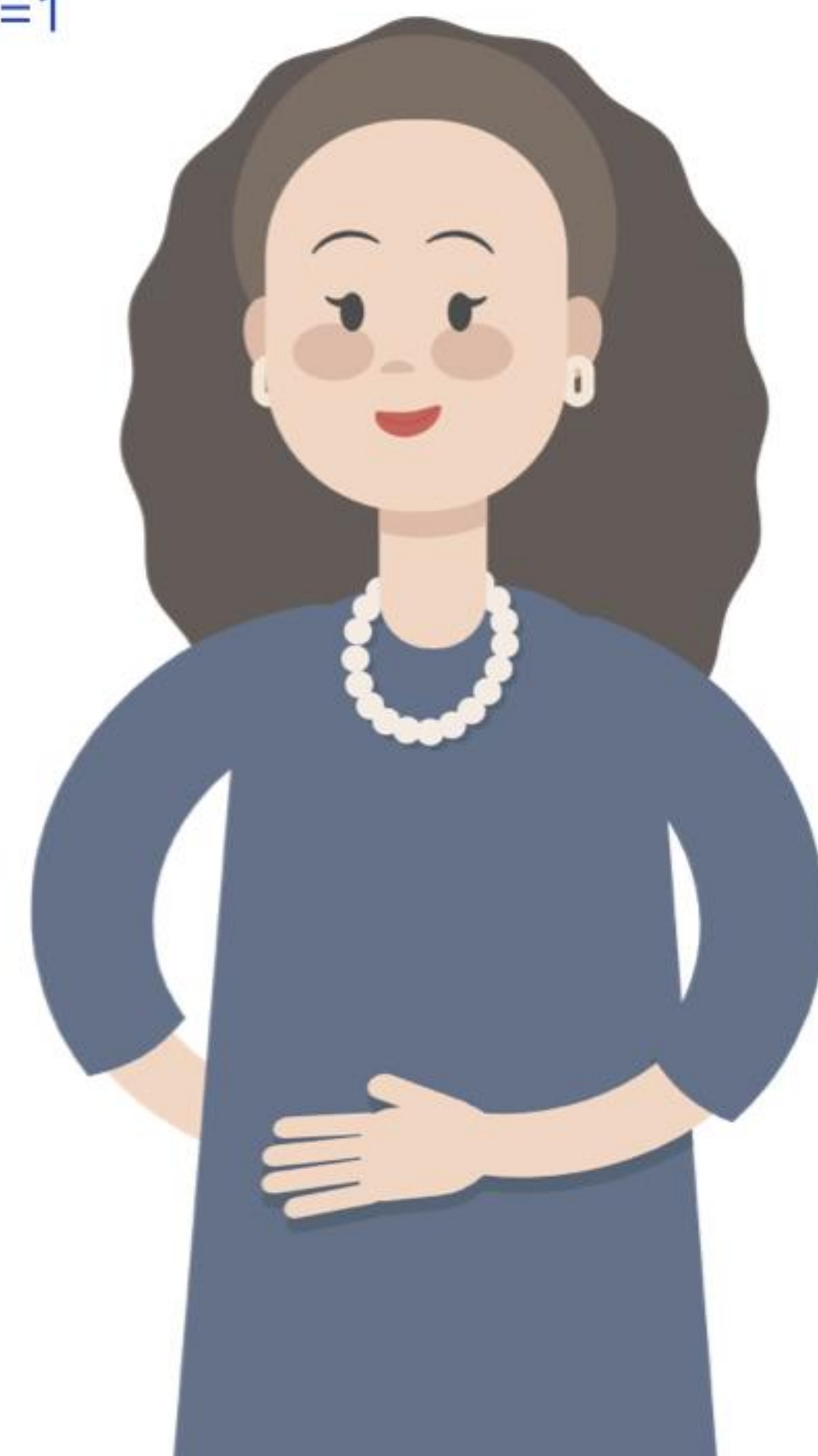
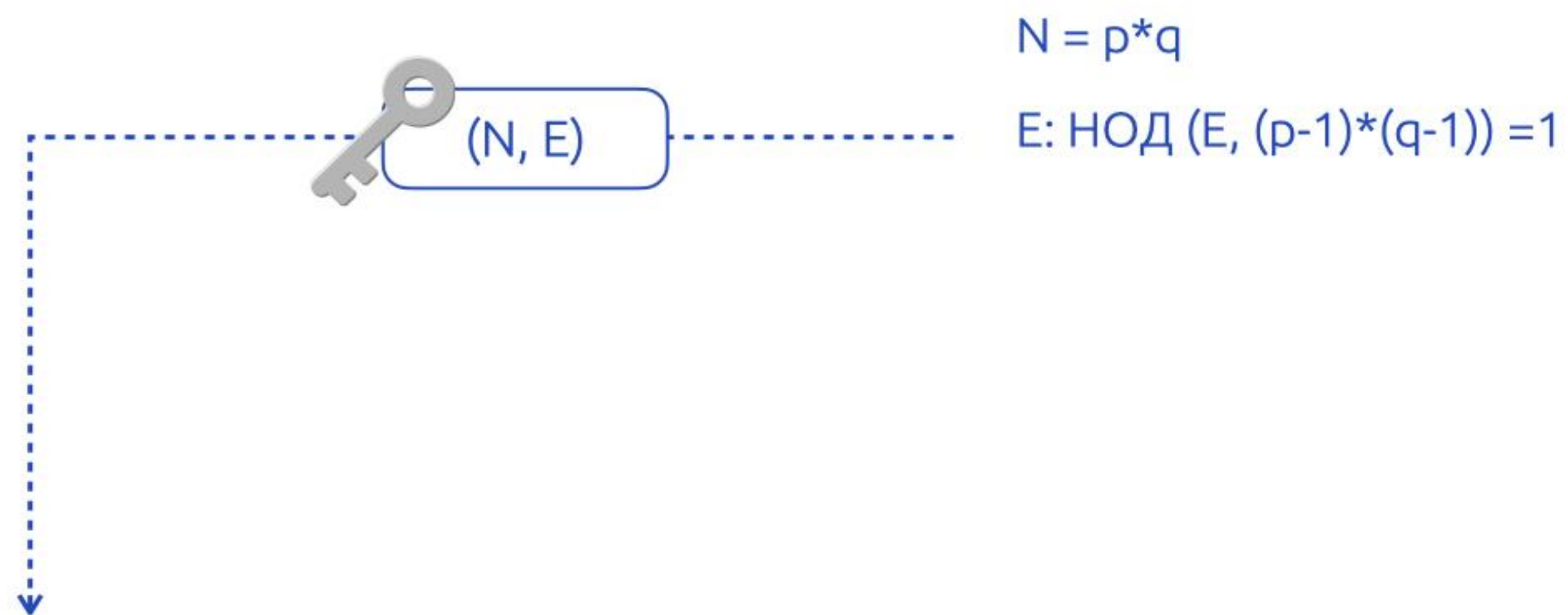
Определение: Криптосистема гомоморфна относительно операции умножения и сложения, если:

$$\begin{aligned} Dec(Enc(k, m_1) \cdot Enc(k, m_2)) &= m_1 \cdot m_2. \\ Dec(Enc(k, m_1) + Enc(k, m_2)) &= m_1 + m_2. \end{aligned}$$

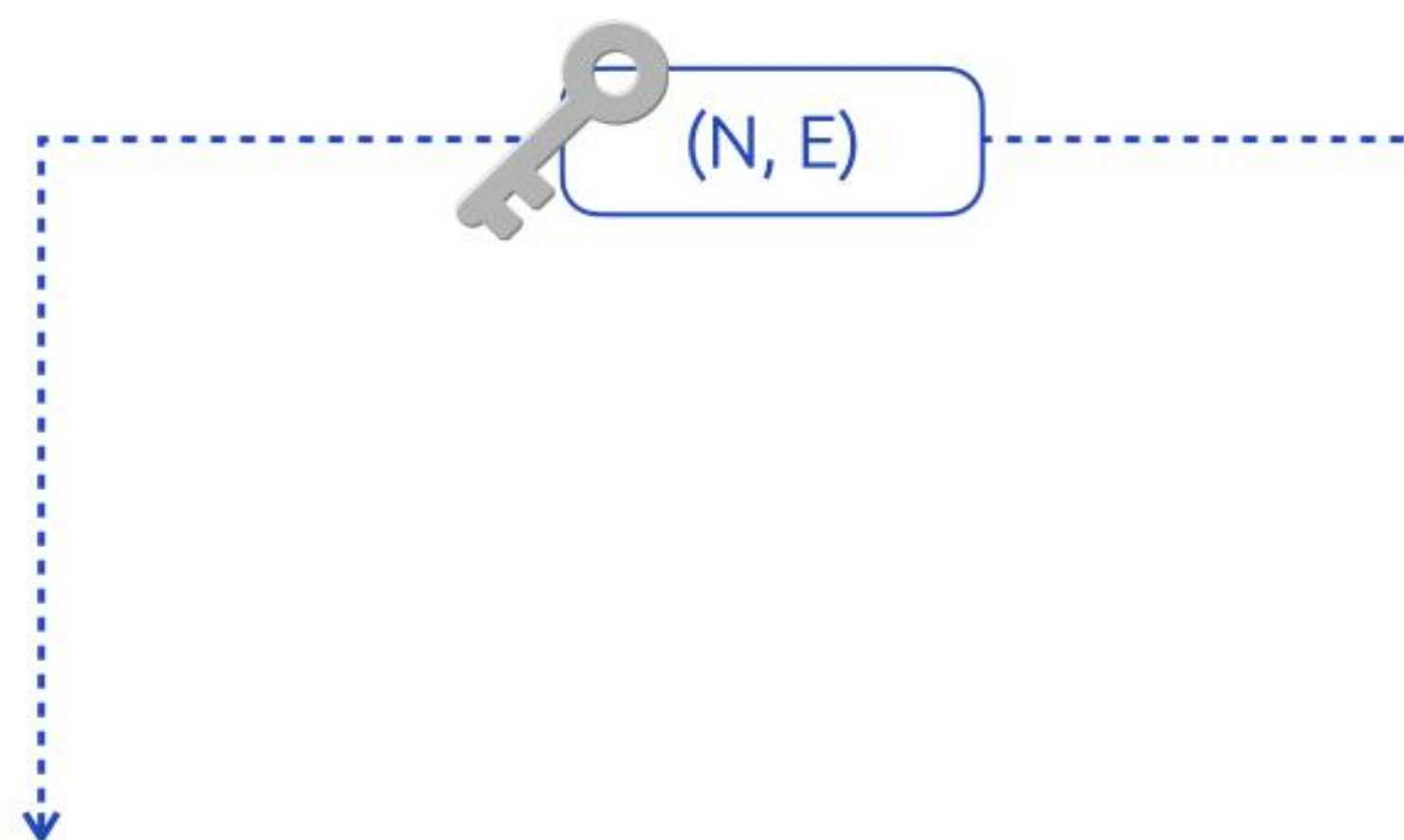
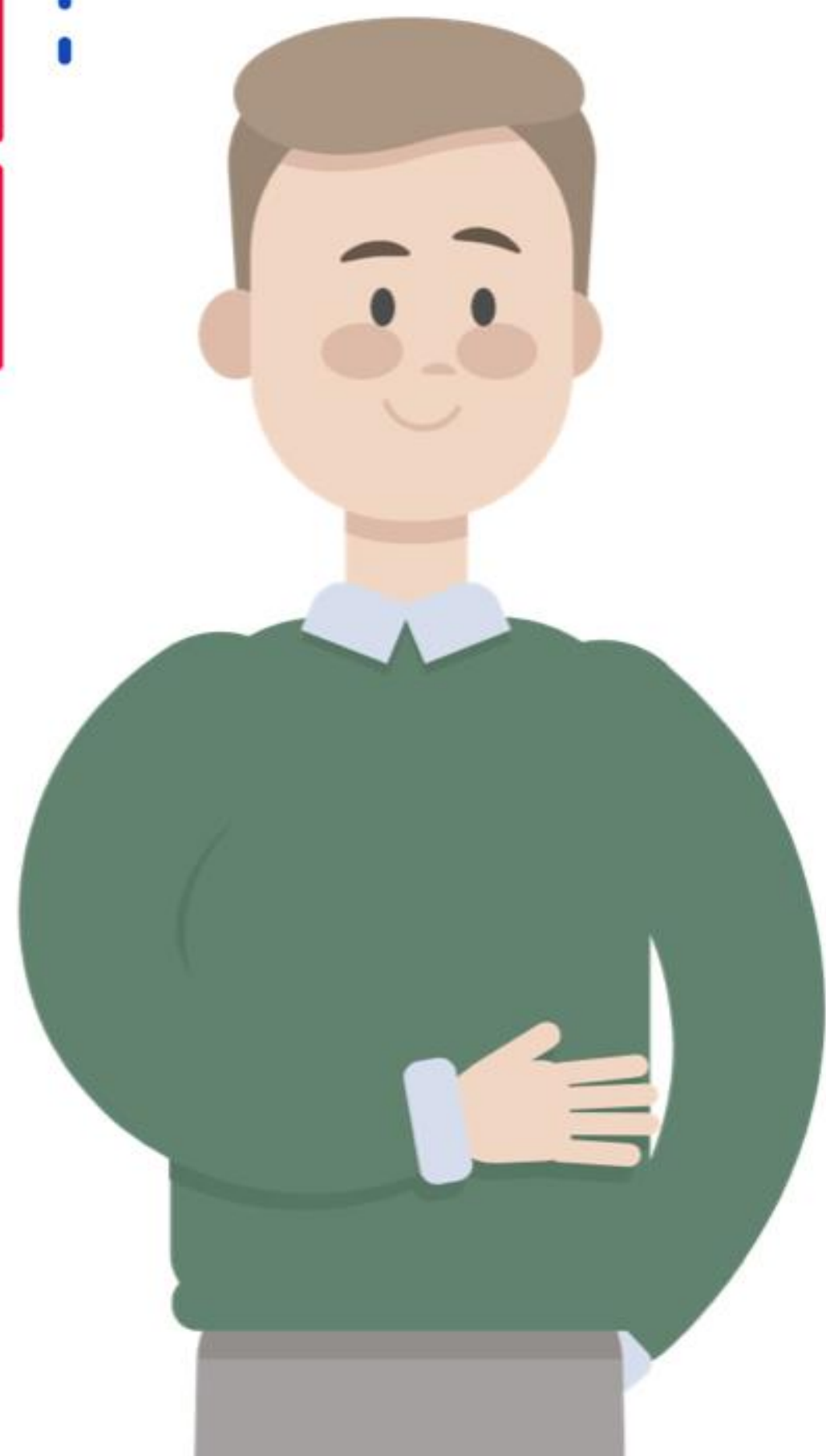
RSA



RSA



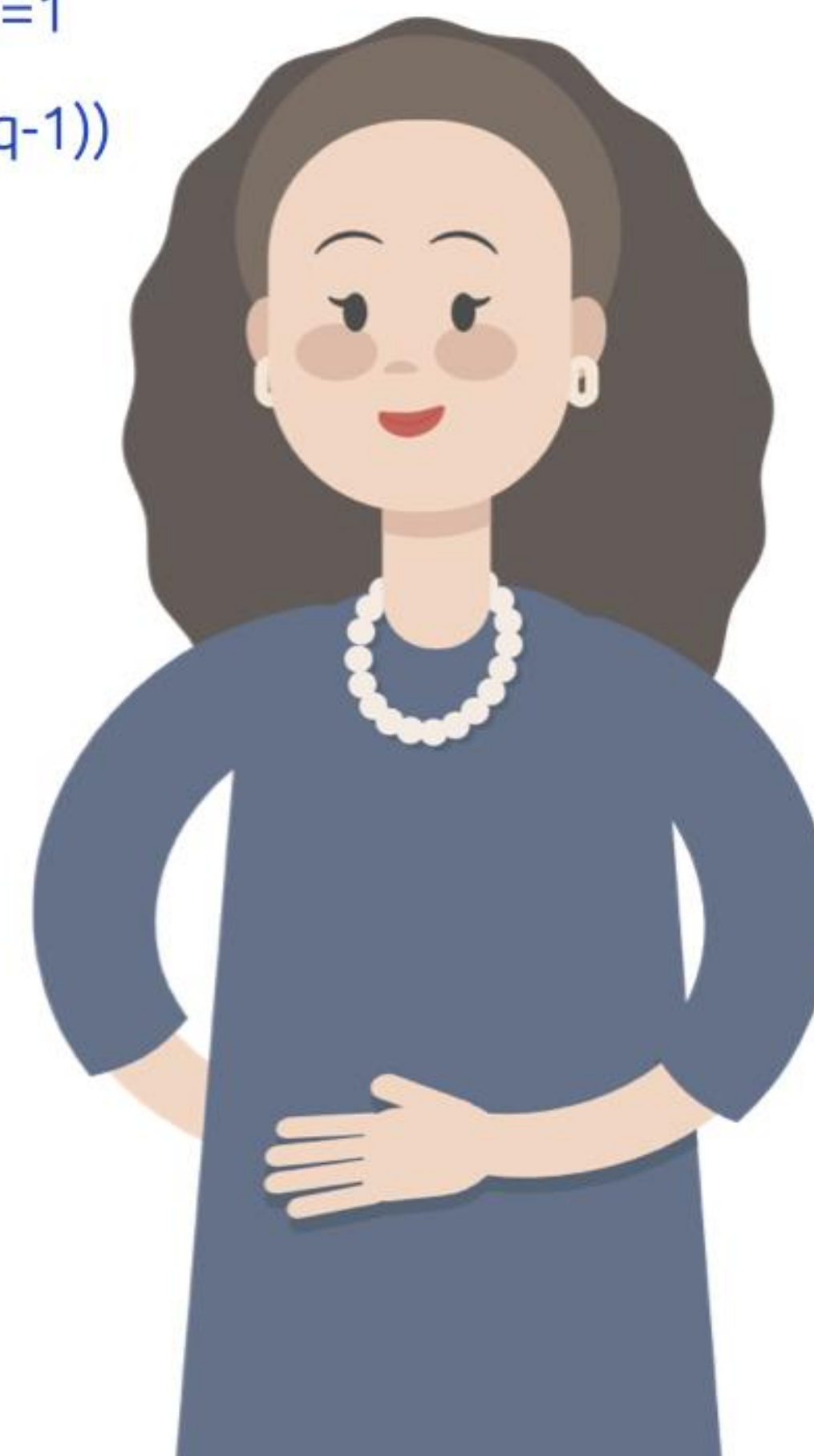
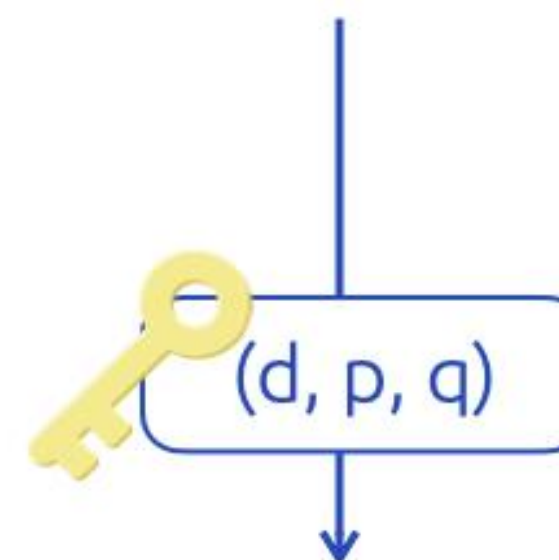
RSA



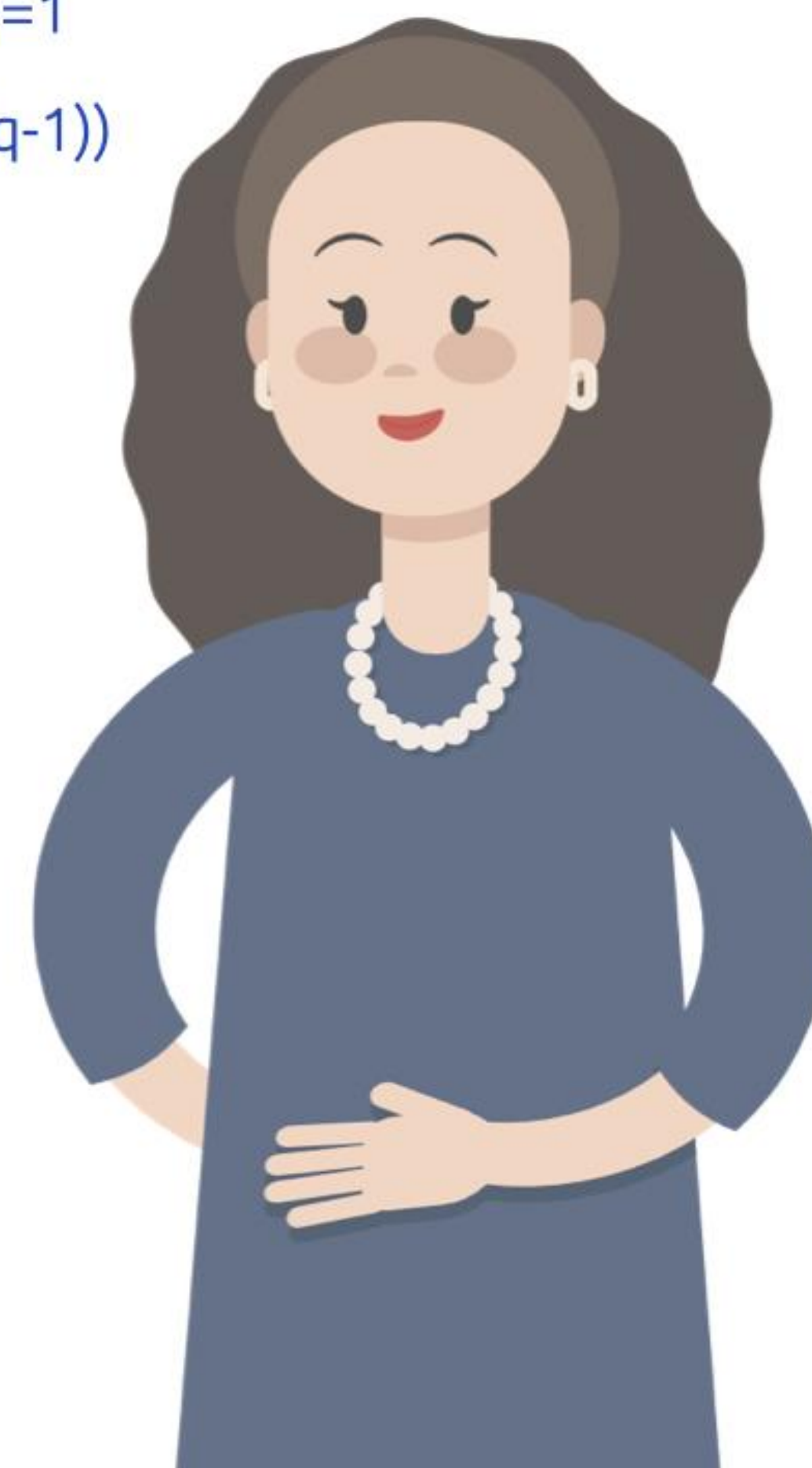
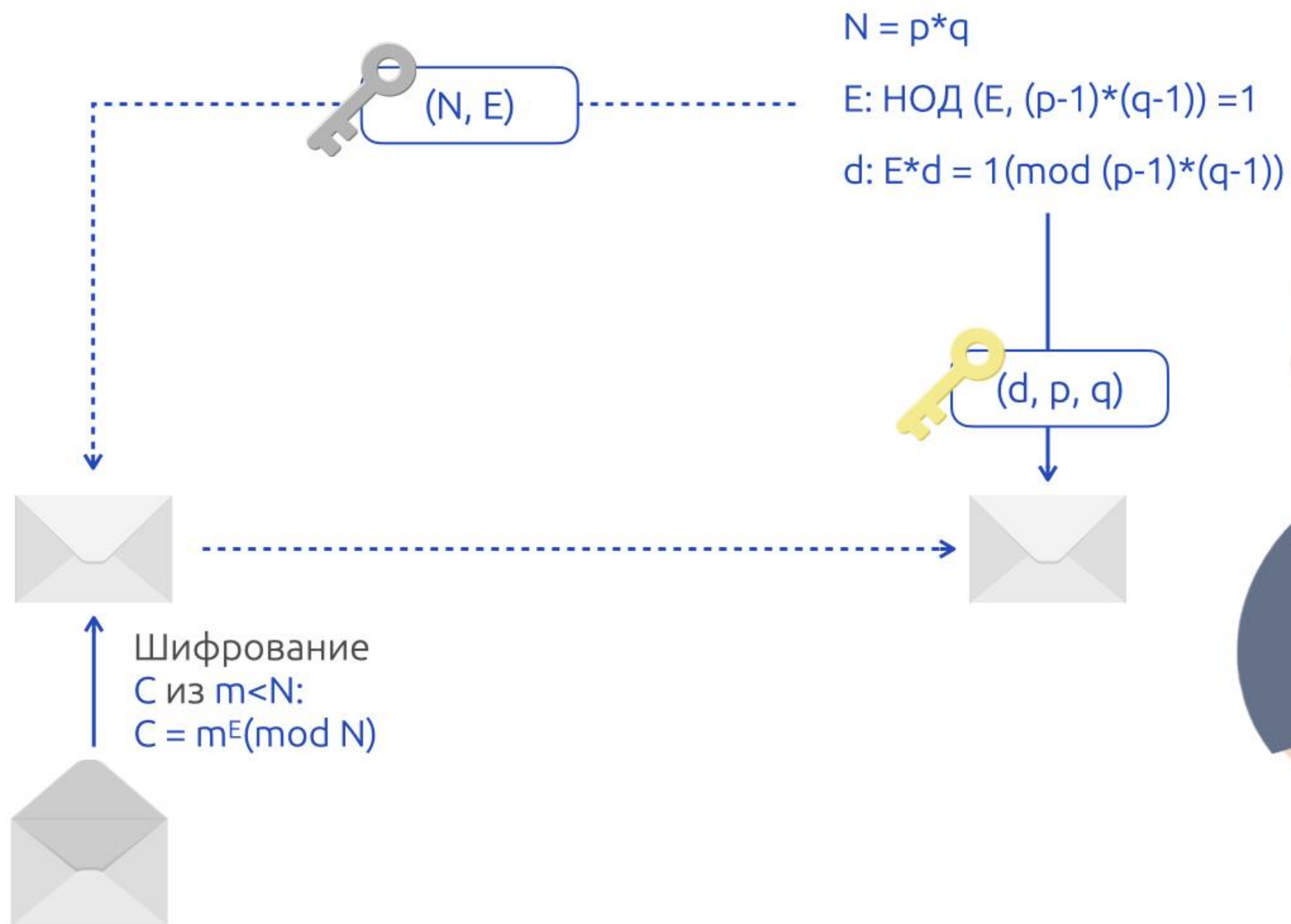
$$N = p * q$$

$$E: \text{НОД}(E, (p-1)*(q-1)) = 1$$

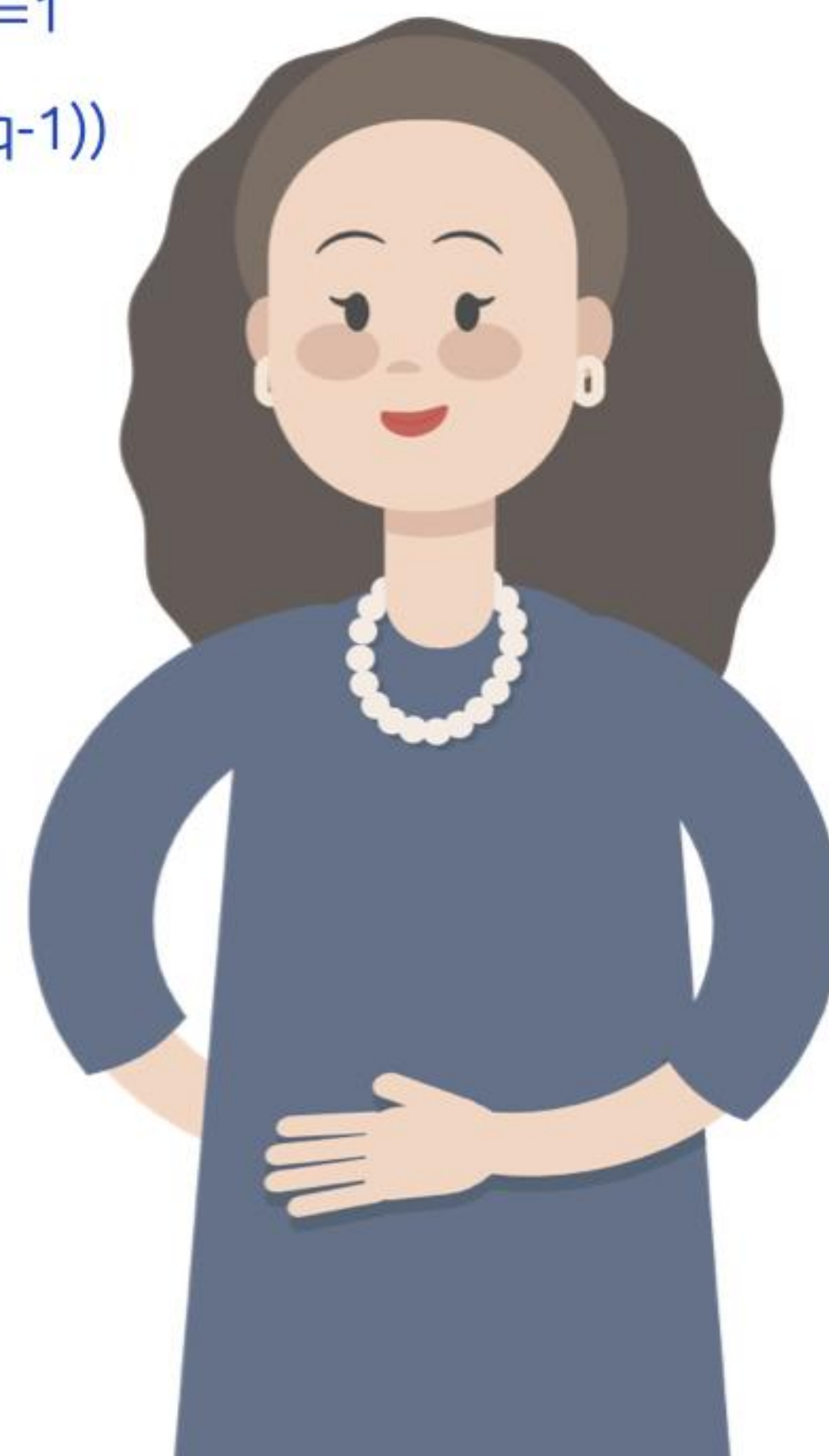
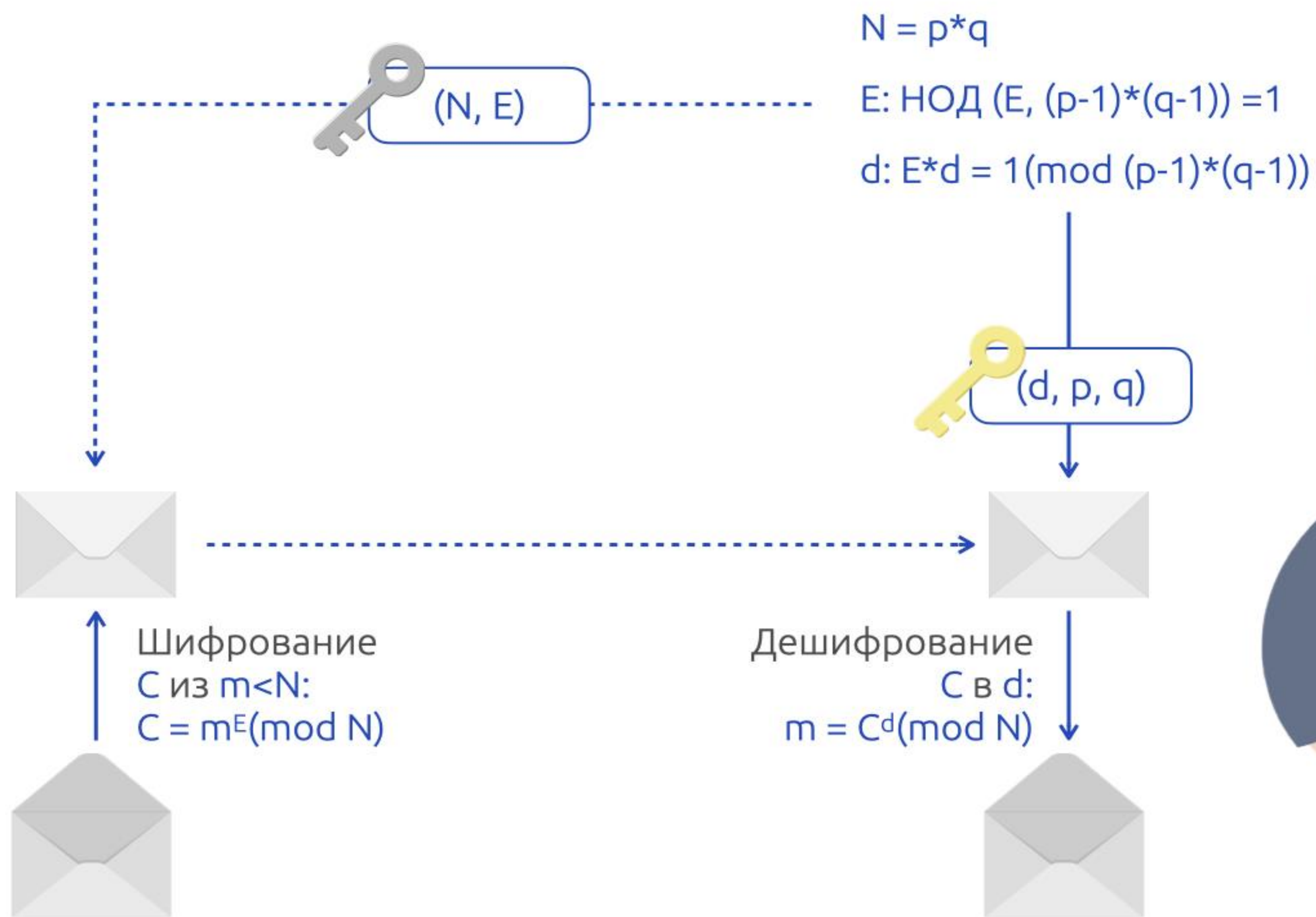
$$d: E * d = 1 \pmod{(p-1)*(q-1)}$$



RSA



RSA



RSA

RSA гомоморфна по умножению:

(N, E) – открытый ключ

m_1, m_2 - открытый текст (шифруемое сообщение)

Enc – шифрующая функция

$$Enc(m_1) \cdot Enc(m_2) = m_1^E \bmod N \cdot m_2^E \bmod N = (m_1 \cdot m_2)^E \bmod N = Enc(m_1 \cdot m_2)$$





Криптосистема Эль-Гамаль



Тахир Эль-Гамаль

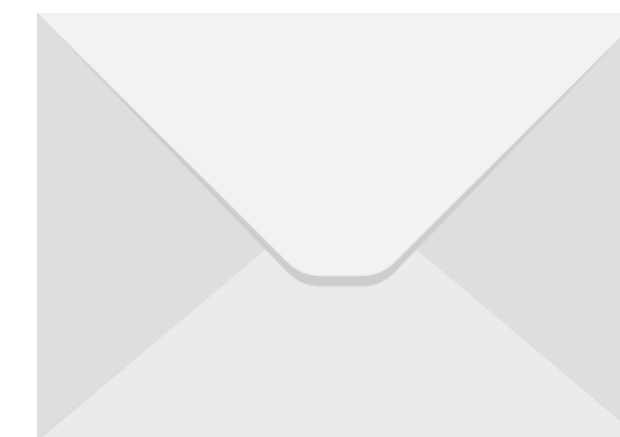
Криптосистема Эль-Гамаль

Параметры домена – параметры которые м.б. использованы многими пользователями.

- P «большое простое число»- 1024 битов, $P-1$ делится на «среднее простое число» Q , лежащее от 2^{160} .
- G – элемент мультипликативной группы поля Z_p^* и $G^{(P-1)/Q} \pmod{P} \neq 1$
- Секретный ключ: $\forall x$ 
- Открытый ключ: $H = G^x \pmod{P}$ 
- Сообщение: не нулевой элемент $m \in Z_p^*$

Шифрование:

- Генерируют случайный эфемерный ключ k
- Вычисляют $C_1 = G^k \pmod{P}$
- Находят $C_2 = m \cdot H^k \pmod{P}$
- Шифротекст $C = (C_1, C_2)$



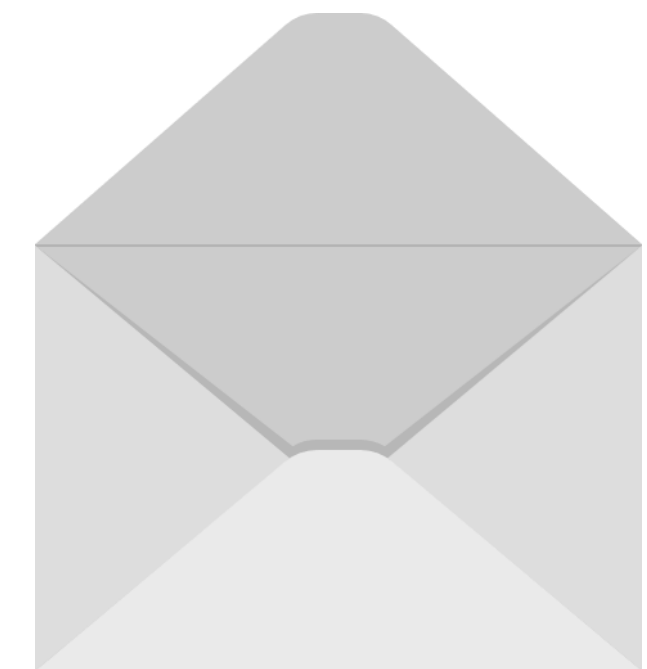
Криптосистема Эль-Гамаль

Дешифрование:

$$(C_2/C_1^x) \pmod{P} = (m \cdot H^k / G^{x \cdot k}) \pmod{P} = (m \cdot G^{x \cdot k} / G^{x \cdot k}) \pmod{P} = m.$$

Применимость:

Подпись GNU Privacy Guard (GnuPG or GPG)



Криптосистема Эль-Гамаль

Вероятностное шифрование - это использование случайности в алгоритме шифрования, так что при шифровании одного и того же сообщения несколько раз оно, как правило, дает разные шифротексты.

Термин "*вероятностное шифрование*" обычно используется в отношении алгоритмов асимметричного шифрования; однако различные алгоритмы симметричного шифрования достигают аналогичного свойства (например, блочные шифры при использовании в режиме цепочки, таком как CBC).

Чтобы быть *семантически безопасным*, то есть скрывать даже частичную информацию о открытом тексте, алгоритм шифрования должен быть вероятностным.

Криптосистема Эль-Гамаль

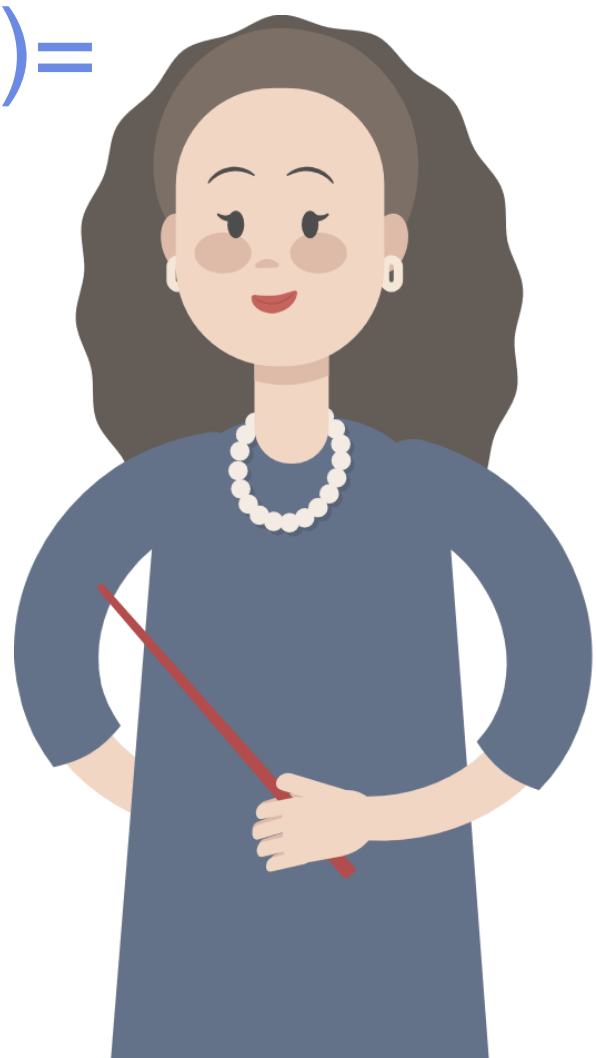
Криптосистема Эль-Гамаль гомоморфна по умножению:

m_1, m_2 - открытый текст (шифруемое сообщение)

Случайный эфемерный ключ для m_1 - k_1 для m_2 - k_2

Enc – шифрующая функция

$$\begin{aligned} Enc(m_1) \cdot Enc(m_2) &= (G^{k_1} \pmod{P}, m_1 H^{k_1} \pmod{P}) \cdot (G^{k_2} \pmod{P}, m_2 H^{k_2} \pmod{P}) = \\ &= (G^{k_1 \cdot k_2} \pmod{P}, (m_1 \cdot m_2) H^{k_1 \cdot k_2} \pmod{P}) = Enc(m_1 \cdot m_2) \end{aligned}$$



Криптосистема Пэ́йе



Криптосистема Пэ́йе

- Секретный ключ: (α, μ, p, q)

$$p, q, \alpha = \text{НаименьшееОбщееКратное}(p-1, q-1),$$

$$\mu = L(g^\alpha \bmod N^2)^{-1} \bmod N,$$

$$L(u) = \text{div} \frac{u-1}{N} \quad (\text{div} - \text{целочисленное деление})$$

- Открытый ключ: (g, N)

$$N = p \cdot q, g - \text{случайное число: } g \in Z_{N^2}^*$$

$Z_{N^2}^*$ - множество целых чисел взаимнопростых с N^2 - это множество состоит из $N \cdot \varphi(N)$ чисел.

- Сообщение: не нулевой элемент $m \in Z_N : m < N$



Криптосистема Пэ́йе (комментарии)

- $\alpha = \text{НаименьшееОбщееКратное}(p - 1, q - 1),$
- $\text{НОД}(p - 1, q - 1)$

$$\alpha = \frac{(p - 1) \cdot (q - 1)}{\text{НОД}(p - 1, q - 1)}$$

- g — случайное число: $g \in Z^*_{N^2}$

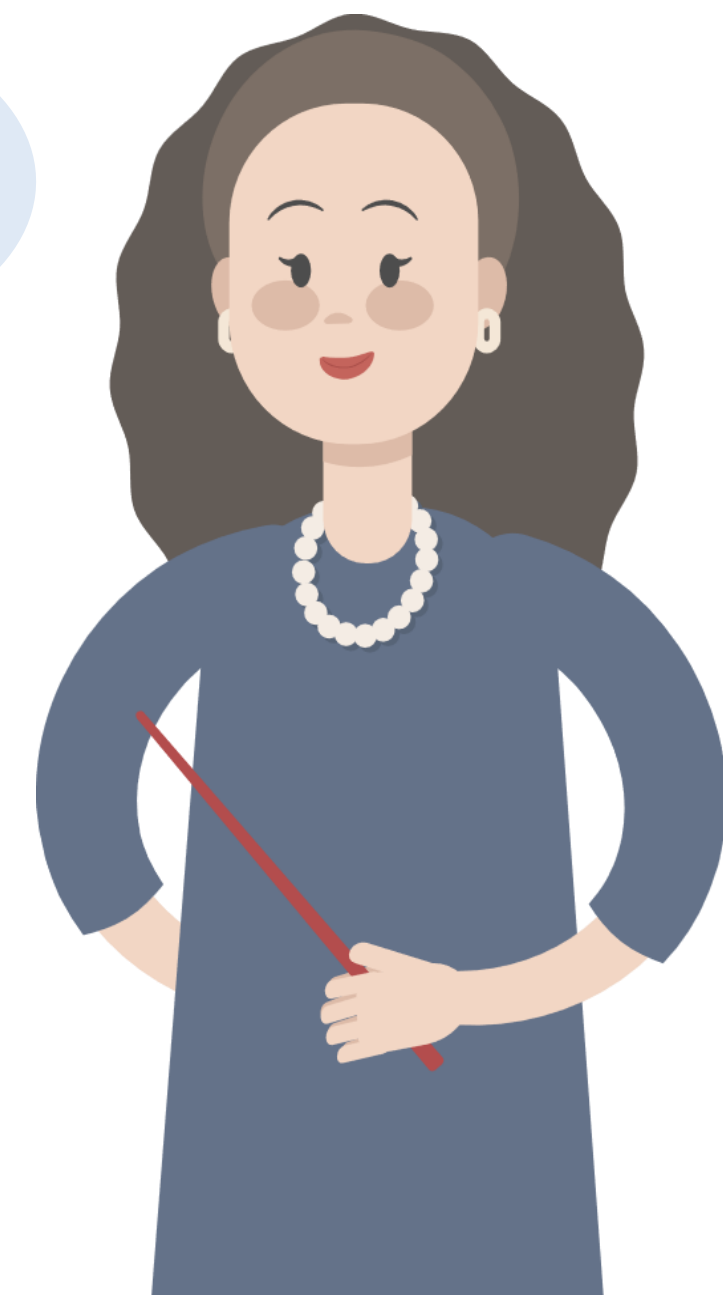
$$\text{НОД}\left(\frac{g^\alpha \bmod N^2 - 1}{N}, N\right) = 1 \quad (1)$$

???

Случайно выбрать α и β из множества $Z^*_{N^2}$, затем вычислить

$$g = (\alpha \cdot N + 1) \cdot \beta^N \bmod N^2$$

В этом случае выбранное g всегда удовлетворяет условию (1).



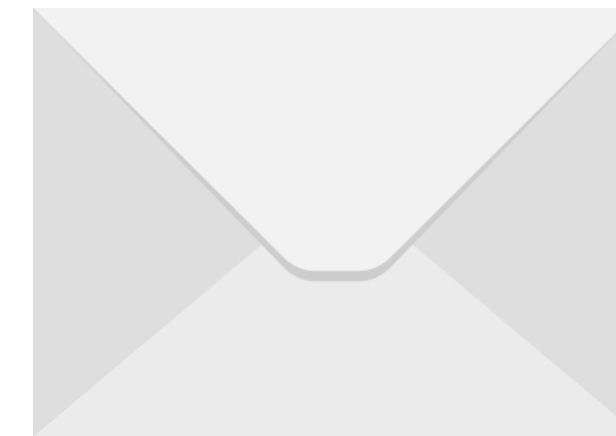
Криптосистема Пэ́йе

Шифрование:

- Генерируют случайное число $r \in Z_N^*$
- $C = g^m \cdot r^N \pmod{N}$

Дешифрование:

- $m = L(C^\alpha \pmod{N^2}) \cdot \mu \pmod{N}$



Пример

1. Генерация ключей

$p = 7$ и $q = 5$, $N = 7 \cdot 5 = 35$, $N^2 = 1225$ и $\alpha = \text{НОК}(6, 4) = 12$.

Выбираем случайное целое число g , такое что $g \in Z_{N^2}^*$, $g = 3$.

Находим $\mu = (L(g^\alpha \bmod N^2))^{-1} \bmod N = 29$.

$(\alpha, \mu, p, q) = (12, 29, 7, 5)$ – закрытый ключ.



Пример

2. Шифрование

- $m=8$
- Выбираем произвольное $r \in Z_N^*$, $r = 9$,
- Вычисляем:

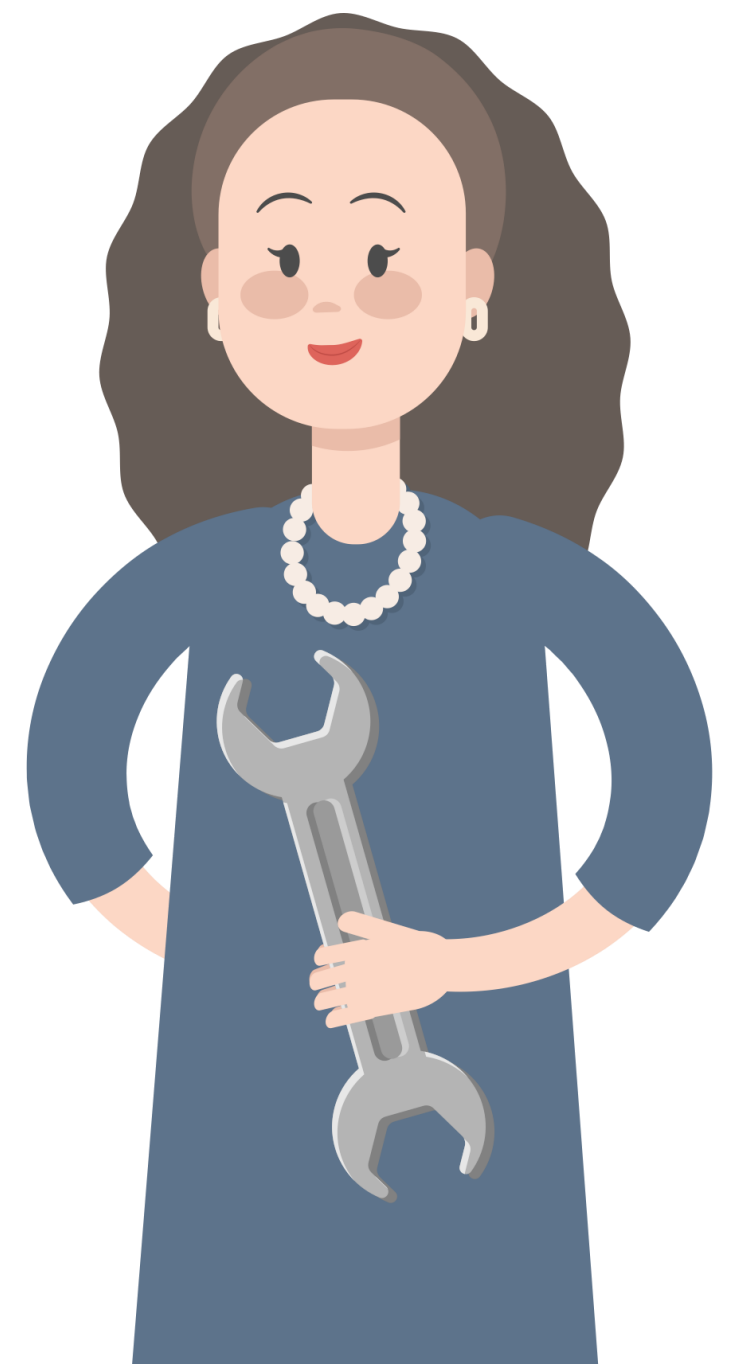
$$C = g^m \cdot r^N \bmod N^2 = 3^8 \cdot 9^{35} \bmod 1225 = 436 \cdot 949 \bmod 1225 = 939.$$



Пример

3. Дешифрование

- $C = 939, C \in Z_{1225}$
- Вычисляем $m = L(C^\alpha \bmod N^2) \cdot \mu \bmod N = L(939^{12} \bmod 1225) \cdot 29 \bmod 35 = 22 \cdot 29 \bmod 35 = 8.$



Криптосистема Пэ́йе

Криптосистема Пэ́йе гомоморфна по сложению:

1. При дешифровании произведения двух шифротекстов будет получена сумма соответствующих им открытым текстам:

- $Dec(Enc(m_1) \cdot Enc(m_2) \bmod N^2) = (m_1 + m_2) \bmod N;$

Частный случай $Dec(Enc(m_1) \cdot g^{m_2} \bmod N^2) = (m_1 + m_2) \bmod N;$

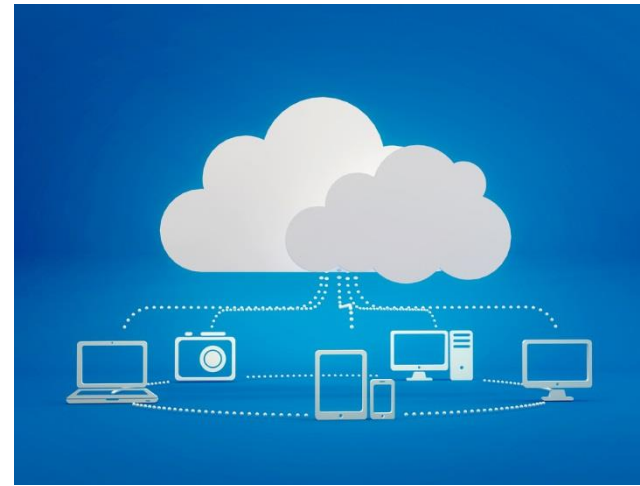
2. При дешифровании криптограммы, возведенной в степень $d \in Z_n^*$, будет получено произведение открытого текста и показателя степени d :

- $Dec(Enc(m)^d) \bmod N^2 = d \cdot m \bmod N.$

Частный случай $Dec(Enc(m_1))^{m_2} \bmod N^2 = m_1 \cdot m_2 \bmod N.$

Применение

1. Облачные вычисления:



Важна производительность, следует применять различные алгоритмы, в зависимости от поставленной задачи.

2. Электронное голосование:



Система сможет зашифровать голоса избирателей и провести расчёты над зашифрованными данными, сохраняя анонимность избирателей.

3. Защищённый поиск информации:



Можно предоставить пользователям возможность извлечения информации из поисковых систем с сохранением конфиденциальности: сервисы смогут получать и обрабатывать запросы, а также выдавать результаты обработки, не зная содержание.

Применение

4. Защита беспроводных децентрализованных сетей связи

Беспроводные децентрализованные самоорганизующиеся сети ([MANET](#)) - сети, состоящие из мобильных устройств.

Для решения проблемы обеспечения безопасности может применяться гомоморфное шифрование, которое встраивается в протоколы маршрутизации для повышения безопасности.

5. Аутсорсинговые услуги для смарт-карт

Универсальные карты с собственной операционной системой, могут выполнять разнообразные функции и взаимодействовать с несколькими поставщиками услуг → некоторые приложения могут работать вне карты на гомоморфно зашифрованных данных.

6. Системы с обратной связью

Системы помогают осуществлять анонимный сбор данных (например опросы)



7. Обфускация для защиты программных продуктов

Компьютерные архитектуры используют двоичные строки, применяя полностью гомоморфное шифрование над битами → можно вычислить любую функцию можно гомоморфно зашифровать целиком всю программу.

Применение

Электронное голосование:

1. Участник схемы разделяет свой голос (секрет) на составляющие (частичные секреты) по соответствующей схеме разделения секрета со свойством гомоморфности по сложению и посылает частичные секреты выборным представителям;
2. Представители складывают полученные голоса; схема гомоморфна по сложению, следовательно, суммы голосов являются частичными секретами соответствующего итога выборов;
3. Доверительное лицо вычисляет конечный итог голосования, используя набор частичных сумм голосов, переданный ему выборными представителями.



Применение

Электронное голосование:

n - кандидатов;

$K=(k_{enc}, k_{dec})$;

Бюллетень: $(p_1, p_2, \dots p_i, \dots p_n)$ - p_i - i -ый кандидат.

Голосование:

Избиратель: $(v_1, v_2, \dots v_i, \dots v_n)$ - $v \in \{0,1\}$

Шифрует ключом k_{enc}

Инициатор: Складывает все результаты и дешифрует ключом k_{dec}



Схема разделения секрета Шамира

Для интерполяции многочлена степени $k-1$ требуется k точек.

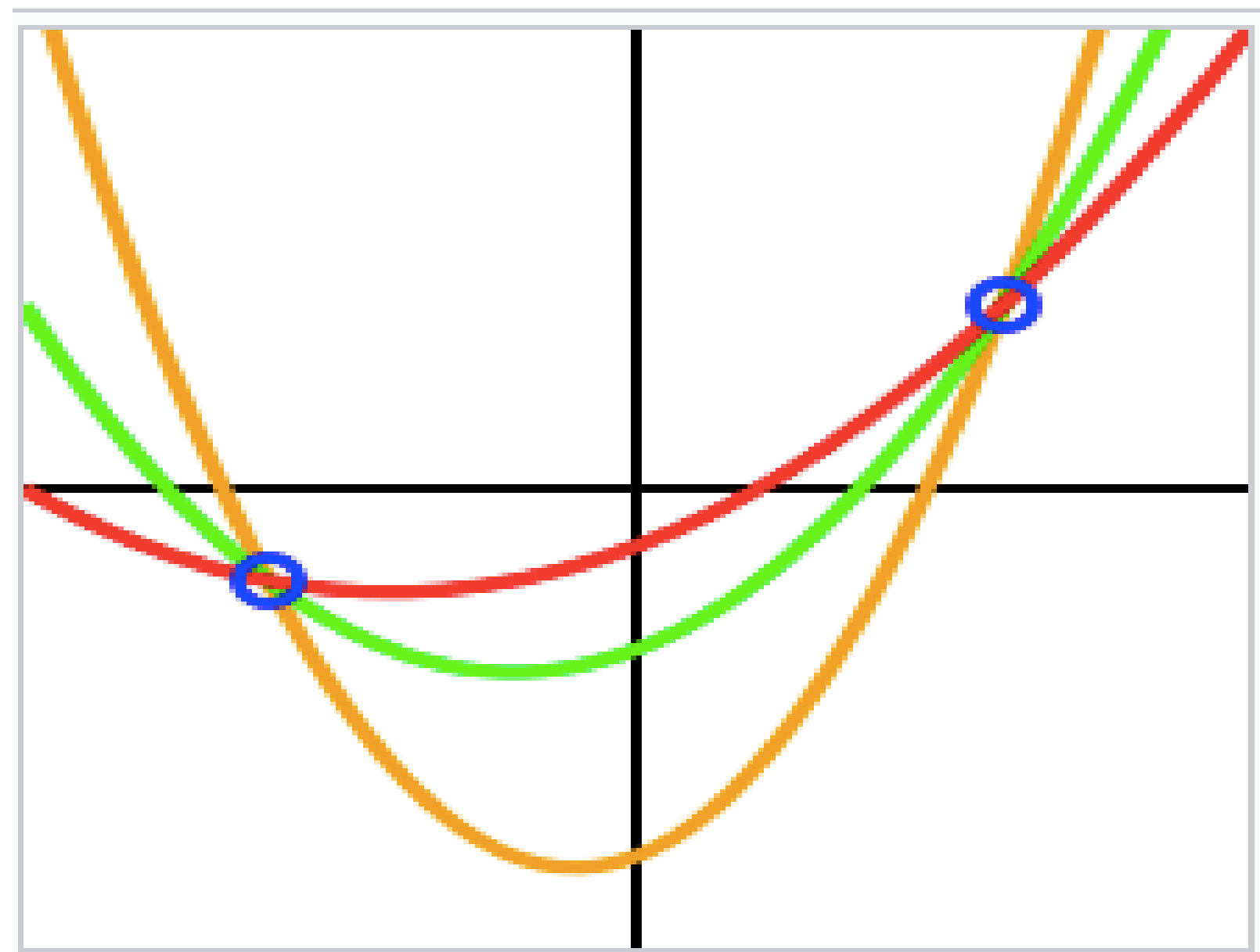


Схема разделения секрета Шамира

1. Подготовка:

M - секрет, k – участников, (n, k) – пороговая схема.

p - простое число $p > M$.

$$F(x) = (a_{k-1}x^{k-1} + a_{k-2}x^{k-2} + \dots + a_1x + M) \mod p$$

2. Создание «долей» секрета:

$$k_1 = F(1) = (a_{k-1} \cdot 1^{k-1} + a_{k-2} \cdot 1^{k-2} + \dots + a_1 \cdot 1 + M) \mod p$$

$$k_2 = F(2) = (a_{k-1} \cdot 2^{k-1} + a_{k-2} \cdot 2^{k-2} + \dots + a_1 \cdot 2 + M) \mod p$$

...

$$k_i = F(i) = (a_{k-1} \cdot i^{k-1} + a_{k-2} \cdot i^{k-2} + \dots + a_1 \cdot i + M) \mod p$$

...

$$k_n = F(n) = (a_{k-1} \cdot n^{k-1} + a_{k-2} \cdot n^{k-2} + \dots + a_1 \cdot n + M) \mod p$$

Схема разделения секрета Шамира

3. Восстановление секрета:

$$F(x) = \sum_i l_i(x) y_i \mod p$$

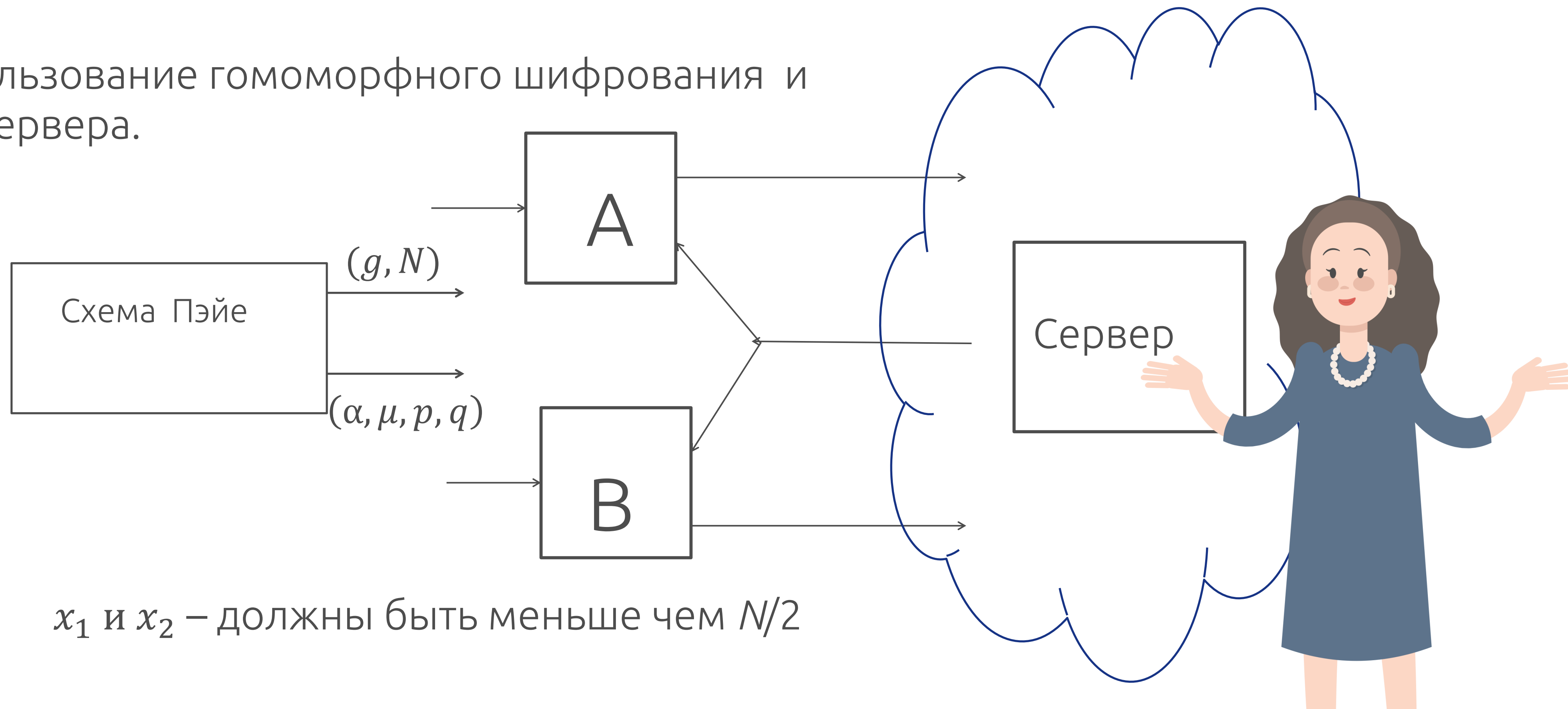
$$l_i(x) = \prod_{i \neq j} \frac{x - x_j}{x_i - x_j} \mod p$$

Применение

8. Анонимные вычисления

Постановка задачи: А и В имеют числа x_1 и x_2 и хотят выяснить у кого число больше, не раскрывая самих значений этих чисел.

Идея – использование гомоморфного шифрования и внешнего сервера.



Применение

1. Пользователь А шифрует число x_1 по схеме Пэйн:

$$C_1 = g^{x_1} \cdot r^N \pmod{N}$$

2. Пользователь В шифрует число x_2 по схеме Пэйн:

$$C_2 = g^{x_2} \cdot r^N \pmod{N}$$

3. Сервер выполняет преобразование зашифрованных данных

$$C = C_1 \cdot C_2^{N-1} \cdot g^l$$

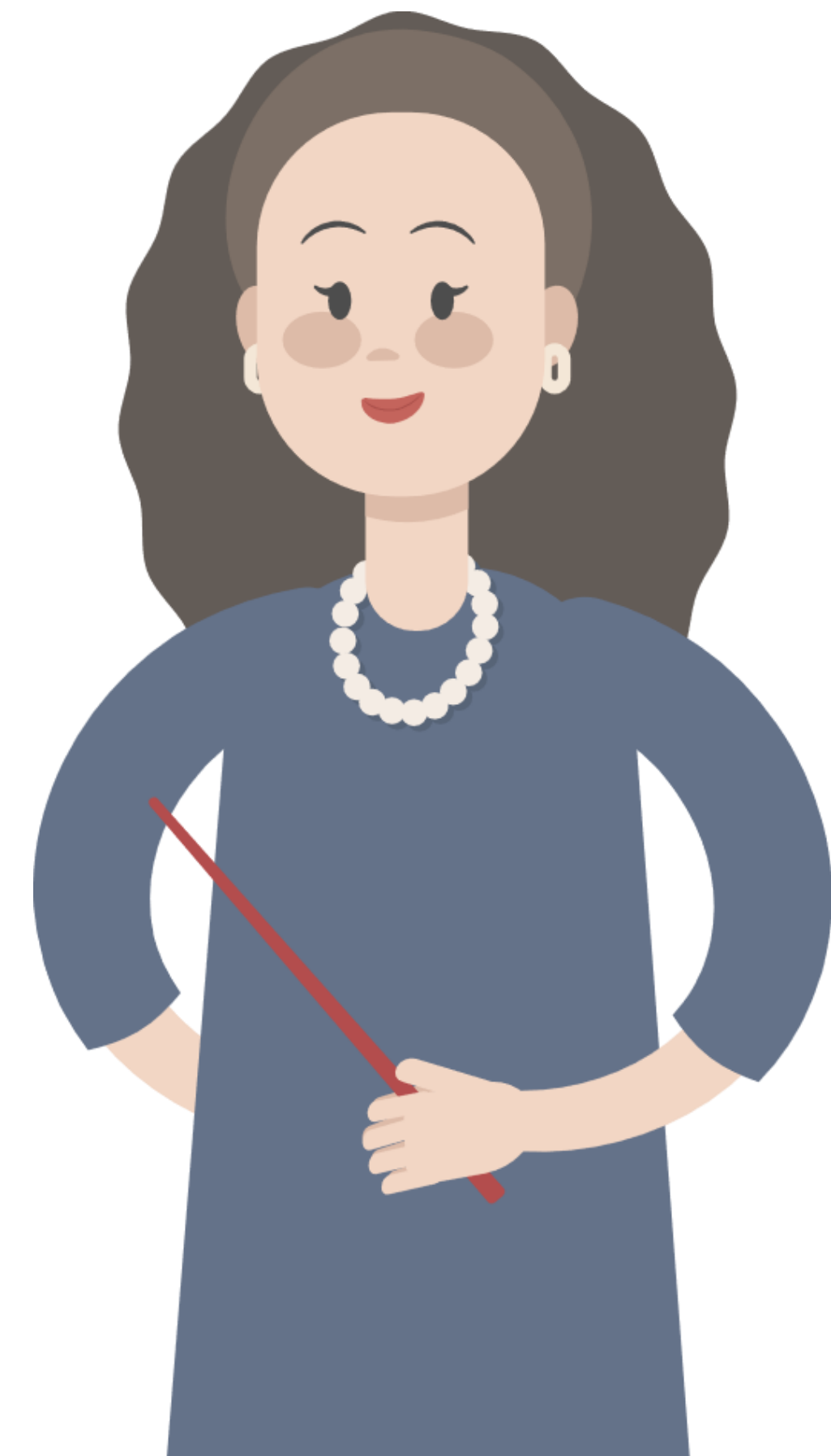
где $l > 0$ - случайное число и отправляет C пользователям.

4. Пользователи А и В дешифруют C и по свойству гомоморфности получают:

$$Dec(C) = L(C^\alpha \pmod{N^2}) \cdot \mu \pmod{N}$$

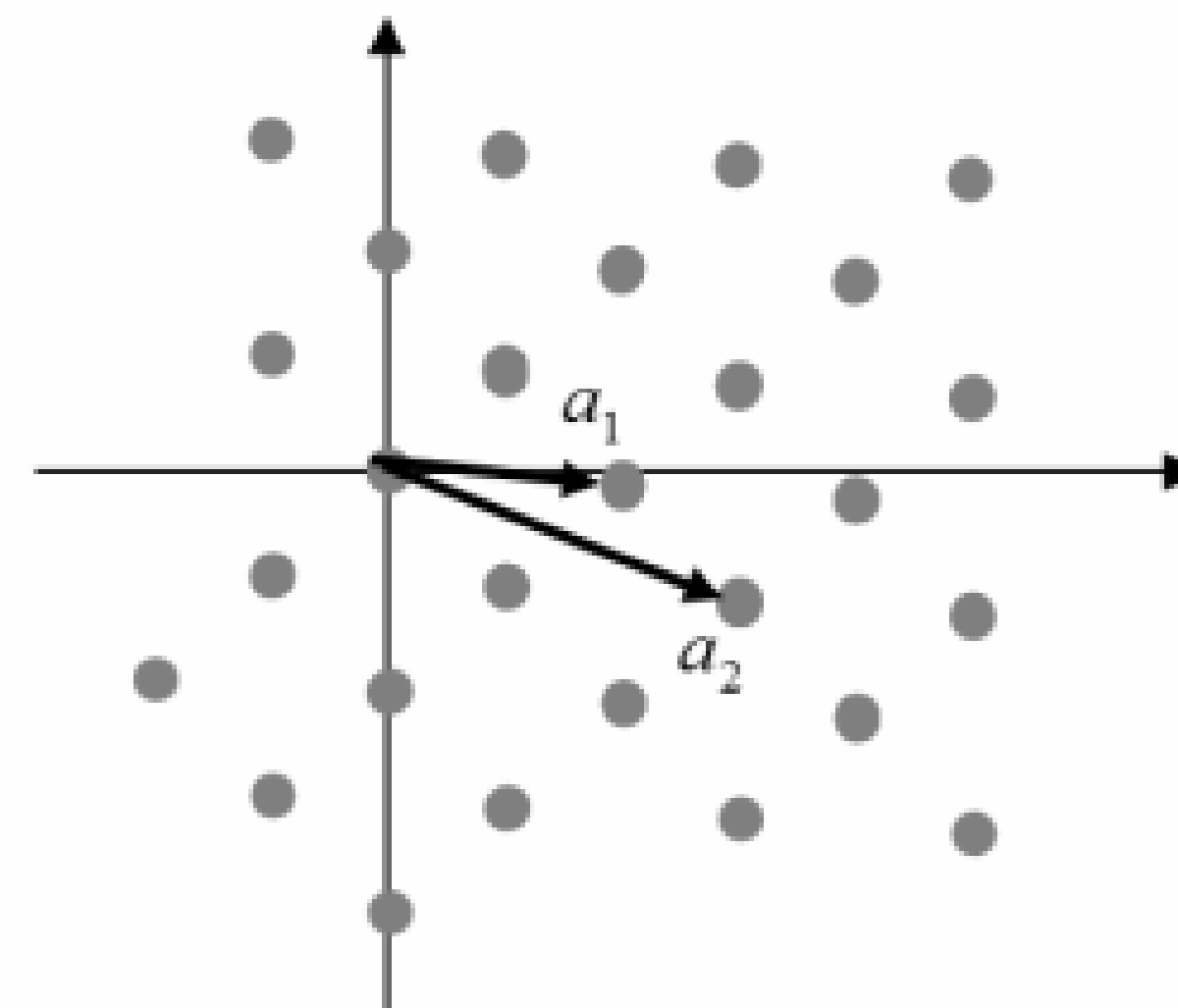
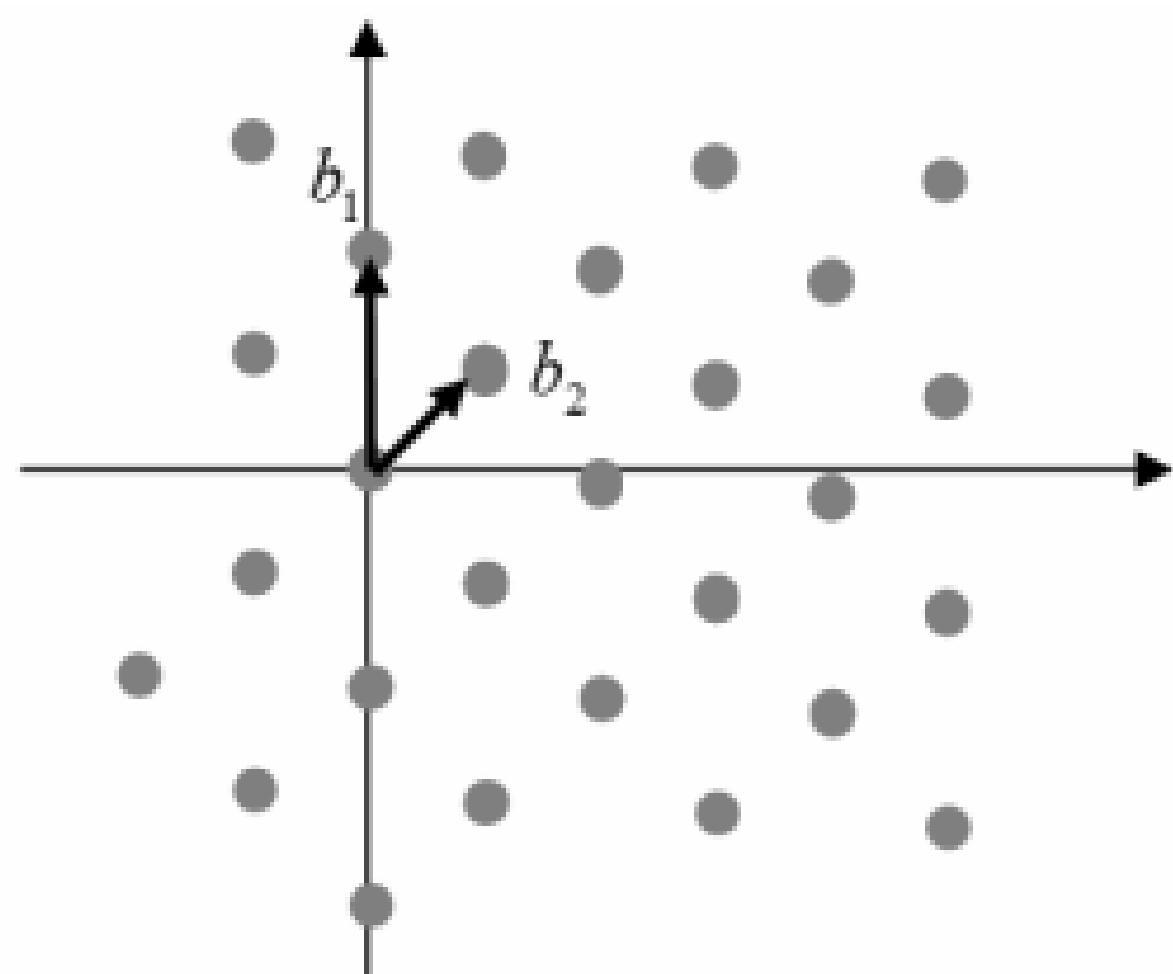
По свойству гомоморфности: $(x_1 + (N-1) \cdot x_2 + l) \pmod{N} = (x_1 - x_2 + l) \pmod{N}$

Если $Dec(C) > \frac{N}{2}$, то $x_1 > x_2$, else $x_1 < x_2$



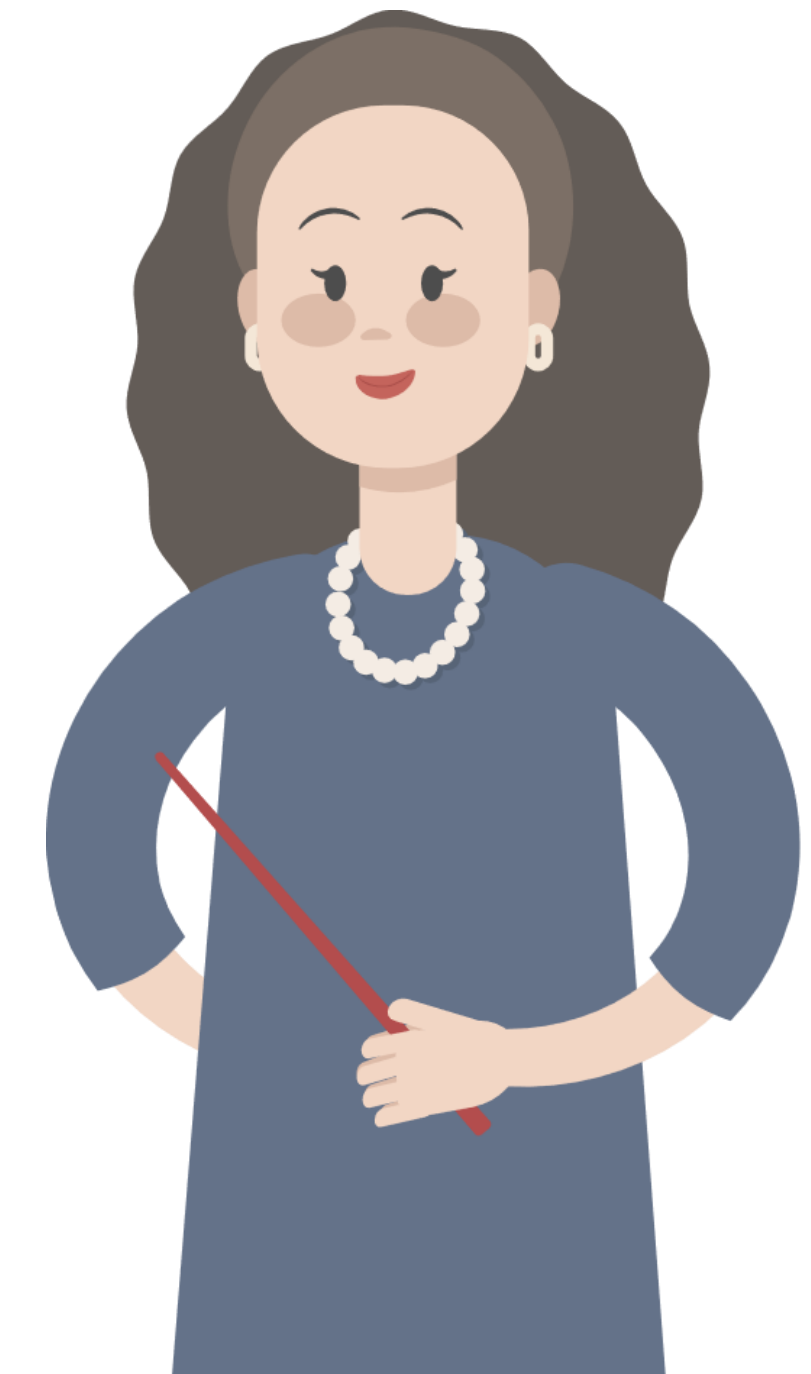
Криптосистема Джентри

Решетка – дискретная аддитивная подгруппа, заданная на множестве R^n , решетку L можно представить как множество векторов заданных целочисленными линейно независимыми базисными векторами $B = \{b_1, \dots, b_n\} \in R^n$, определенными по модулю некоторого целого числа $x \in Z^n$, $L = \sum_{i=1}^n b_i Z = \{bx : x \in Z^n\}$. У решетки может быть множество базисов, $L = \sum_{i=1}^n a_i Z$



Криптосистема Джентри

- Задача нахождения кратчайшего вектора (*Shortest Vector Problem*) — найти в заданном базисе решётки ненулевой вектор по отношению к определённой нормали.
- Задача нахождения ближайшего вектора (*Closest Vector Problem*) — нахождение вектора в решётке по заданному базису и некоторому вектору, не принадлежащему решётке, при этом максимально схожего по длине с заданным вектором.



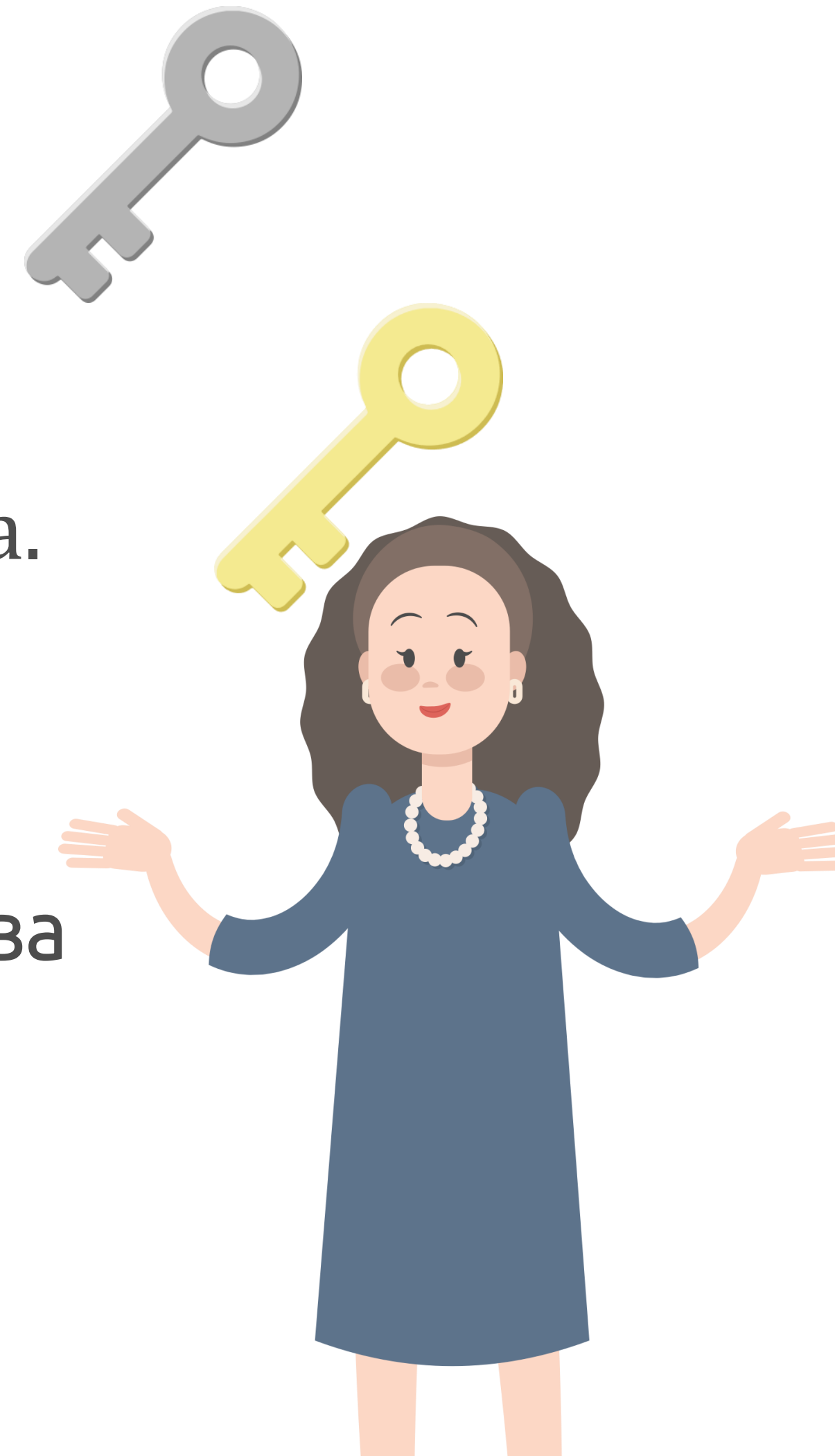
Криптосистема Джентри

- Генерация ключа:

1. Выбирается N ;
2. Выбирается секретный ключ $k_{sec} \ll N$, $\text{НОД}(k_{sec}, N) = 1$;
3. Выбирается открытый ключ k_{open} — набор чисел a_i :
$$a_i = r \cdot k_{sec} + e \cdot N$$
, где e и r — случайные числа.

- Шифрование:

Шифротекст будет являться суммой произвольного количества элементов открытого ключа и открытого текста.



Криптосистема Джендри

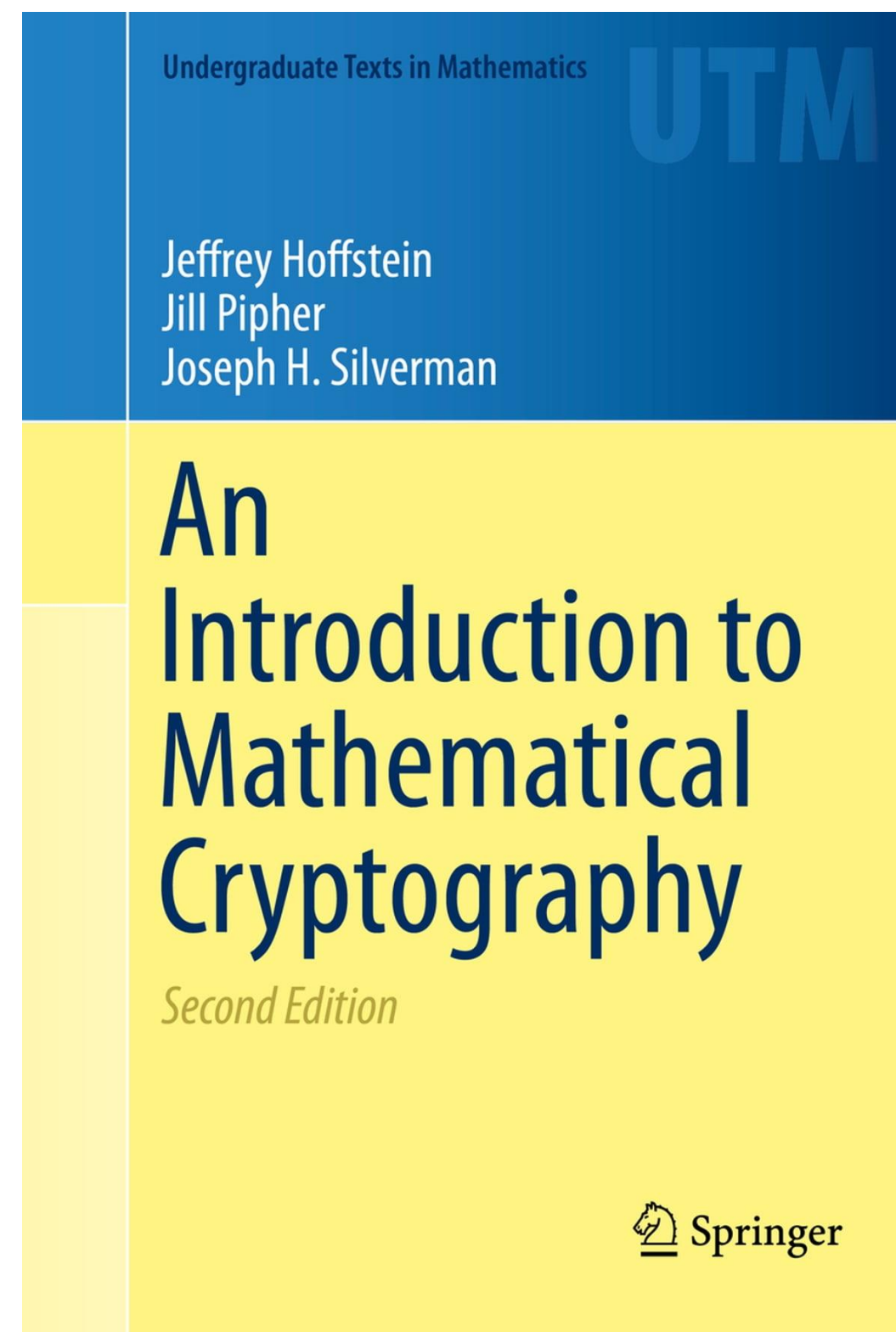
Гомомρφность по сложению:


$$Dec(Enc(m_1) + Enc(m_2)) = Dec(\sum_{i=0}^N m_{1,i} \cdot a_i + \sum_{i=0}^N m_{2,i} \cdot a_i) = Dec(Enc(m_1 + m_2))$$



NTRU

NTRUEncrypt, изначально называвшийся NTRU, был изобретён в 1996 году математиками Jeffrey Hoffstein, Jill Pipher и Joseph H. Silverman, разработавшие систему вместе с основателем компании NTRU Cryptosystems, Inc. Daniel Lieman.





Попробуем
завоевать
криптомир!

Что мы будем
делать сегодня?