(возвести в квадрат и умножить). Последовательность операций определяется следующим образом: если очередное двоичное число в бинарном разложении равно 0, то применяется операция ^, а иначе — операция ^×. В нашем примере последовательность такова: (^×)(^×)(^)(^×). Эта последовательность операций для базового числа 7 записывается следующим образом:

$$\Big(\Big(\big((1\hat{}2\times 7)\hat{}2\times 7\big)\hat{}2\Big)\hat{}2\times 7\Big) = \Big(\Big(\big((1^2\times 7)^2\times 7\big)^2\Big)^2\times 7\Big).$$

Теперь для каждой скобки (начиная с самой внутренней) выполняем соответствующую операцию по mod 10, используя для каждого последующего шага результат предыдущего вычисления:

$$1^2\times 7 \bmod 10 = 7,$$
$$7^2\times 7 \bmod 10 = 343 \bmod 10 = 3,$$
$$3^2 \bmod 10 = 9,$$
$$9^2\times 7 \bmod 10 = 567 \bmod 10 = 7.$$

8.1. Бинарное представление. Рассмотренный выше метод возвести в квадрат и умножить начинается с получения бинарного представления степени. Рассмотрим два самых простых способа перевода десятичного числа в бинарное представление. Принцип работы каждого из них рассмотрим на примере перевода десятичного числа $x = 15$ в двоичное.

Начать со старшей цифры. Обозначим $x_0 = x$. Находим наибольшую степень m_0 двойки, для которой $2^{m_0} \le x$. Ясно, что $m_0 = 3$, так как $2^3 \le 15 < 2^4$. Повторим эту же процедуру, но для числа $x_1 \stackrel{\text{def}}{=} x_0 - 2^{m_0} = 15 - 2^3 = 7$. Получаем число $m_1 = 2$; вычисляем $x_2 = x_1 - 2^{m_1} = 7 - 2^2 = 3$.

Далее действуем по этому же принципу: находим $m_2 = 1$; вычисляем $x_3 = x_2 - 2^{m_2} = 2 - 2^1 = 1$; находим $m_3 = 0$; вычисляем $x_4 \stackrel{\text{def}}{=} x_3 - 2^{m_3}$. В общем случае алгоритм заканчивает работу, когда очередное x становится равным 0.

Чтобы записать двоичное представление числа 15, необходимы $m_0 + 1$ позиций: самая левая позиция имеет номер 0, а самая правая — номер m_0. В позициях m_0, m_1, m_2, ... этого представления записываем единицы, а других позициях — нули. Итак, $15 = 1111_2$.

Начать с младшей цифры. Число $x_0 = 15$ нечетное, поэтому $b_0 = 1$ (иначе следовало бы выбрать $b_0 = 0$). Пусть $x_1 \stackrel{\text{def}}{=} (x_0 - b_0)/2$. Число $x_1 = 7$ нечетное, поэтому $b_1 = 1$. Продолжаем в том же духе: $x_2 \stackrel{\text{def}}{=} (x_1 - b_1)/2 = 3$, $b_2 = 1$, $x_3 \stackrel{\text{def}}{=} (x_2 - b_2)/2 = 1$, $b_3 = 1$, $x_4 \stackrel{\text{def}}{=} (x_3 - b_3)/2 = 0$. Алгоритм заканчивает работу, когда очередное x становится равным 0. В нашем примере алгоритм закончился на x_4, поэтому требуется 4 позиции, чтобы записать двоичное представление числа 15: $15 = b_3 b_2 b_1 b_0 = 1111_2$.

Преимущество этого алгоритма в том, что не требуется предварительного вычисления степеней двойки, но зато приходится неоднократно выполнять операцию деления на 2, которая віполняется довольно “медленно”.

9. Как выбрать число, взаимно простое с θ

Для работы метода RSA необходимо найти число b, взаимно простое с θ. Иными словами, необходимо найти число b, для которого $\gcd(b, \theta) = 1$. Для проверки последнего условия используют алгоритм Евклида, который позволяет находить наибольший общий делитель двух чисел.

Алгоритм выбора b является рандомизированным.

Шаг 1. Сгенерировать случайное число $b < \theta$
Шаг 2. С помощью алгоритма Евклида найти $\gcd(b, \theta)$
Шаг 3. **if** $\gcd(b, \theta) = 1$ **then stop**
else повторить шаг 1.

Алгоритм 4. Выбор числа b

Быстродействие алгоритма 4 определяется функцией

$$\varphi(n) \stackrel{\text{def}}{=} \text{количество чисел} < n, \text{ взаимно простых с } n,$$

которую называют *функцией Эйлера*. Вероятность того, что на отдельном шаге будет выбрано взаимно простое с θ число, примерно равна $\frac{\varphi(\theta)}{\theta}$. Поэтому для нахождения b в среднем требуется

$$\frac{\theta}{\varphi(\theta)} \tag{12}$$

шагов алгоритма 4.[1] При "удачном" выборе p и q (напомним, что $\theta = (p-1)(q-1)$) отношение $\frac{\varphi(\theta)}{\theta}$ может быть близко к 1 и поэтому алгоритм 4 закончится за несколько шагов.

С другой стороны, "неудачный" выбор p и q приводит к отношению $\frac{\varphi(\theta)}{\theta}$ близкому к 0 и поэтому алгоритм 4 может выполняется достаточно длительное время.

Среди существующих оценок для функции Эйлера отметим следующую, которая выполняется для составных $\theta > 6$:

$$\sqrt{\theta} < \varphi(\theta) < \theta - \sqrt{\theta}.$$

Ясно, что $\varphi(p) = p - 1$, если p — простое число.

[1] Время ожидания "*успеха*" — это геометрическая случайная величина с параметром $\frac{\varphi(\theta)}{\theta}$. Математическое ожидание такой случайной величины равно число в (12).

10. Алгоритм Евклида

Этот алгоритм описан дважды в “Началах” Евклида. Упоминание об алгоритме имеется в более ранних источниках, поэтому Евклид по-видимому не является его изобретателем.

Покажем, как найти $\gcd(1071, 462)$ с помощью алгоритма Евклида. Для начала, от 1071 отнимем такое кратное значение 462, чтобы разница былы меньше, чем 462:

$$1071 = 2 \times 462 + 147.$$

Затем от 462 отнимем такое кратное значение 147, чтобы разница была меньше, чем 147:

$$462 = 3 \times 147 + 21.$$

Затем от 147 отнимем такое кратное значение 21, чтобы разница была меньше, чем 21:

$$147 = 7 \times 21 + 0.$$

Так как последний остаток равен нулю, алгоритм заканчивается и $\gcd(1071, 462) = 21$.

10.1. Общий случай. Пусть $r_0 > r_1$ два натуральных числа. Описанный ниже алгоритм Евклида требует $O(\log_2 r_0)$ делений.

Input: $x < y$
Output: $\gcd(x, y)$
Шаг 1. $r_0 \stackrel{\text{def}}{=} y$, $r_1 \stackrel{\text{def}}{=} x$; $k = 1$
Шаг 2. **if** r_{k-1} делится на r_k **then** $\gcd(x, y) = r_k$ **stop**
else $k := k + 1$, повторить Шаг 2.

Алгоритм 5. Алгоритм Евклида

В алгоритме 5 рассмотривается последовательность r_k, $k \geq 0$, образованная по правилу:

Шаг 1: если r_0 делится на r_1, то последовательность $\{r_k\}$ состоит из двух членов, r_0 и r_1; в ином случае, мы вычисляем $r_2 = r_0 \bmod r_1$;

Шаг 2: если формирование последовательности $\{r_k\}$ не закончено и r_1 делится на r_2, то последовательность $\{r_k\}$ состоит из трех членов, r_0, r_1 и r_2; в противном случае, вычисляем $r_3 = r_1 \bmod r_2$ и повторяем шаг 2 для пары r_2 и r_3.

Процедура продолжается до тех пор, пока не получим пару, в которой одно из чисел делится на другое. Такая пара будет найдена за конечное количество шагов, так как члены последовательности $\{r_k\}$ монотонно убывают и неотрицательны:

Обозначим последний индекс у членов последовательности $\{r_k\}$ через n. Тогда

$$
\begin{aligned}
&r_0 = k_1 r_1 + r_2, \qquad 0 < r_2 < r_1,\\
&r_1 = k_2 r_2 + r_3, \qquad 0 < r_3 < r_2,\\
&r_2 = k_3 r_3 + r_4, \qquad 0 < r_4 < r_3,\\
&\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\\
&r_{n-2} = k_{n-1} r_{n-1} + r_n, \qquad 0 < r_n < r_{n-1},\\
&r_{n-1} = k_n r_n.
\end{aligned}
$$

Пусть $\alpha \stackrel{\text{def}}{=} \gcd(r_0, r_1)$ — наибольший общий делитель чисел r_0 и r_1. Тогда $\alpha = \gcd(r_1, r_2)$. Действительно, так как и r_0, и r_1 делятся на α, то и r_2 делится на α. Если $\gcd(r_1, r_2) = \gamma\alpha$, то b r_1, и r_2 делятся на $\gamma\alpha$, по'тому и r_0 делится на $\gamma\alpha$. Отсюда вытекает, что $\gamma = 1$ и следовательно $\alpha = \gcd(r_1, r_2)$.

Аналогично этому доказывается, что $\alpha = \gcd(r_{k-1}, r_k)$ для любого $1 \leq k \leq n$. Отсюда мы заключаем, что $\alpha = r_n$.

Действительно, r_n делится на α. Если $r_n = \gamma\alpha$, то на $\gamma\alpha$ делится и r_{n-1}, откуда сразу вытекает, что $\gamma = 1$, так как $\alpha = \gcd(r_{n-1}, r_n)$.

11. Расширенный алгоритм Евклида

В алгоритме 3 необходимо найти число a, для которого $ab \equiv 1 \pmod{\theta}$ при заданных b и θ. Такое число a называется обратным для b по модулю θ.

Обратное по модулю существует не для всех пар b и θ, но в важном случае $\gcd(b, \theta) = 1$ это действительно так.

Теорема 1. *Пусть u и v натуральные числа, причем $\gcd(u, v) = d$. Тогда существует пара целых чисел x и y, для которых*

$$xu + yv = d. \tag{13}$$

Доказательство теоремы 1 основано на так называемом *расширенном алгоритме Евклида*. Сначала мы выведем из теоремы 1 результат, необходимый для алгоритма 3, а потом докажем и саму теорему 1.

Следствие 1. *Пусть b и θ — натуральные числа, причем $\gcd(b, \theta) = 1$. Тогда существует обратное к b по модулю θ.*

Доказательство следствия 1. Согласно теореме 1 существуют целые x и y, при которых $xb + y\theta = 1$. Поэтому $1 = (xb + y\theta) \bmod \theta = xb \bmod \theta$, то есть x и есть обратное к b по модулю θ. □

Доказательство теоремы 1. Равенство (13) очевидно при $u = v$, так как $d = u = v$ (достаточно в (13) выбрать $x = 1$ и $y = 0$).

Рассмотрим случай $u \neq v$. Не теряя общности считаем, что $u > v$. Если $v = 0$, то $\gcd(u, v) = u$ и поэтому равенство (13) выполнено при $x = 1$ и $y = 0$.

Если же $v \neq 0$, то обозначим $q = [u/v]$. Тогда

$$u \bmod v = u - qv.$$

Равенство (13), но для пары $[v, u \bmod v]$ вместо пары $[u, v]$, означает, что существуют целые числа x_1 и y_1, при которых

$$x_1 v + y_1(u \bmod v) = \gcd(v, u \bmod v) \tag{14}$$

или

$$x_1 v + y_1(u - qv) = \gcd(u, v),$$

так как $\gcd(u, v) = \gcd(v, u \bmod v)$. Последнее равенство можно переписать в виде

$$y_1 u + (x_1 - q y_1) v = \gcd(u, v),$$

что равносильно (13) при $x = y_1$ и $y = x_1 - q y_1$. Таким образом из равенства (14) вытекает (13). Заметим, что вторая координата в паре $[v, u \bmod v]$ неотрицательна, но строго меньше второй координаты в паре $[u, v]$.

Повторяя проведенное рассуждение необходимое количество раз, в конце концов получим пару с нулевой второй координатой, для которой представление типа (13) очевидно (как отмечено в начале доказательства). Из представления для последней пары получим представление для предыдущей пары, как описано выше. Продолжая в том же духе, через конечное количество шагов придем к равенству (13). $\square$

УПРАЖНЕНИЯ

Упражнение 1. Доказать, что любой аргумент $0 \leq x \leq 9999$ однозначно восстанавливается по значению функции $f(x) = 12^x \bmod 10^4$.

Упражнение 2. Почему 0 не может принадлежать совокупности (3), если a является примитивным корнем по отношению к n?

10. “Начала” Евклида написаны примерно за 300 лет до нашей эры. Историки математики считают, что алгоритм был известен Евдоксу (375 лет до н. э.). Ван дер Варден доказывал, что этот алгоритм использовался уже в школе Пифагора (570 г.–495 г. до н. э.).

В “Началах” Евклида он описан дважды — в VII книге для нахождения наибольшего общего делителя двух натуральных чисел и в X книге для нахождения наибольшей общей меры двух однородных величин. В обоих случаях дано геометрическое описание алгоритма, для нахождения “общей меры” двух отрезков.