Binary and Matrix Exponentiation

Lecture 6: Number Theory

Rishabh Dhiman

Algorithms and Coding Club
Indian Institute of Technology Delhi

18 July 2021

Problem

Find $a^n \mod m$ *for* $1 \leq a, m \leq 10^9$ *and* $1 \leq n \leq 10^{18}$.

Binary Exponentiation

For $n = 2^k$,

$$a^{2^k} = \left(a^{2^{k-1}}\right)^2$$

```
int b = a;
for (int i = 0; i < k; ++i)
    b = b * b % M;
```

Binary Exponentiation

For other n, $n = (n_{k-1} \ldots n_1 n_0)_2$.

$$a^n = a^{\sum_{i=0}^{k-1} n_i 2^i} = \prod_{i=0}^{k-1} a^{n_i 2^i}.$$

```
int b = a, res = 1;
while (n != 0) {
    if (n & 1)
        res = res * b % M;
    n >>= 1;
    b = b * b % M;
}
```

Binary Exponentiation

For other n, $a = 3$, $n = 13$.

$$13 = (1101)_2.$$

$$\begin{array}{lll} n = (1101)_2 & res = 1 & b = 3 \\ n = (110)_2 & res = 1 \times 3 & b = 9 \\ n = (11)_2 & res = 1 \times 3 & b = 81 \\ n = (1)_2 & res = 1 \times 3 \times 81 & b = 6561 \\ n = 0 & res = 1 \times 3 \times 81 \times 6561 & b = 43046721 \end{array}$$

Generalizing Binary Exponentiation

Only property of multiplication that was used is that it is associative.

Generalizing Binary Exponentiation

Only property of multiplication that was used is that it is associative.
We can replace multiplication with any associative binary operator,

$$(a * b) * c = a * (b * c).$$

Generalizing Binary Exponentiation

Example of such operators,

1. Addition
2. Matrix Multiplication

Problem

Given a number a and n, find

$$\underbrace{a + a + \cdots + a}_{n\ times}.$$

Generalizing Binary Exponentiation

Repeated Addition

```
int b = a, res = 0;
while (n != 0) {
    if (n & 1)
        res = res + b;
    n >>= 1;
    b = b + b;
}
```

Problem

Given a matrix A and a positive integer n, find A^n.

```
Matrix b = a, res = id;
while (n != 0) {
    if (n & 1)
        res = res * b;
    n >>= 1;
    b = b * b;
}
```

Application of Matrix Exponentiation

Problem

Find the n-th Fibonacci number modulo m for $1 \leq n \leq 10^{18}$.

Application of Matrix Exponentiation

$$F_n = F_{n-1} + F_{n-2}$$

Application of Matrix Exponentiation

$$F_n = F_{n-1} + F_{n-2}$$

$$\begin{bmatrix} F_n \\ F_{n-1} \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} F_{n-1} \\ F_{n-2} \end{bmatrix}$$

Application of Matrix Exponentiation

$$
\begin{aligned}
\begin{bmatrix} F_n \\ F_{n-1} \end{bmatrix} &= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} F_{n-1} \\ F_{n-2} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^2 \begin{bmatrix} F_{n-2} \\ F_{n-3} \end{bmatrix} \\
&\vdots \\
&= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^k \begin{bmatrix} F_{n-k} \\ F_{n-k-1} \end{bmatrix} \\
&= \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix}^{n-1} \begin{bmatrix} F_1 \\ F_0 \end{bmatrix}
\end{aligned}
$$

Product-Oriented Recurrence

Codeforces Round #566 Div. 2E

Problem

Compute $f_n \mod 10^9 + 7$ *for* $1 \leq n \leq 10^{18}$ *where,*

$$f_n = c^{2n-6} \cdot f_{n-1} \cdot f_{n-2} \cdot f_{n-3} \text{ for } n \geq 4$$

given f_1, f_2, f_3.

Product-Oriented Recurrence

Codeforces Round #566 Div. 2E

Let $f_n = c^{g_n}$,

Product-Oriented Recurrence
Codeforces Round #566 Div. 2E

Let $f_n = c^{g_n}$,

$$\begin{aligned} c^{g_n} &= c^{2n-6} c^{g_{n-1}} c^{g_{n-2}} c^{g_{n-3}} \\ &= c^{2n-6+g_{n-1}+g_{n-2}+g_{n-3}} \end{aligned}$$

$$\implies g_n = 2n - 6 + g_{n-1} + g_{n-2} + g_{n-3}$$

Product-Oriented Recurrence

Codeforces Round #566 Div. 2E

Let $f_n = c^{g_n}$,

$$\begin{aligned} c^{g_n} &= c^{2n-6} c^{g_{n-1}} c^{g_{n-2}} c^{g_{n-3}} \\ &= c^{2n-6+g_{n-1}+g_{n-2}+g_{n-3}} \\ \implies g_n &= 2n-6+g_{n-1}+g_{n-2}+g_{n-3} \\ \iff g_n + n &= g_{n-1} + (n-1) + g_{n-2} + (n-2) + g_{n-3} + (n-3) \\ \iff h_n &= h_{n-1} + h_{n-2} + h_{n-3} \end{aligned}$$

where you define $h_n = g_n + n$.

Product-Oriented Recurrence

Codeforces Round #566 Div. 2E

$$
\begin{bmatrix} h_n \\ h_{n-1} \\ h_{n-2} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} h_{n-1} \\ h_{n-2} \\ h_{n-3} \end{bmatrix}
$$

$$
= \begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^{n-3} \begin{bmatrix} h_3 \\ h_2 \\ h_1 \end{bmatrix}
$$

Let

$$\begin{bmatrix} 1 & 1 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}^{n-3} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix}.$$

$$\begin{bmatrix} h_n \\ h_{n-1} \\ h_{n-2} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} h_3 \\ h_2 \\ h_1 \end{bmatrix}$$

$$\implies h_n = a_{11}h_3 + a_{12}h_2 + a_{13}h_1.$$

Product-Oriented Recurrence

Codeforces Round #566 Div. 2E

$$\begin{bmatrix} h_n \\ h_{n-1} \\ h_{n-2} \end{bmatrix} = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} h_3 \\ h_2 \\ h_1 \end{bmatrix}$$

$$\implies h_n = a_{11}h_3 + a_{12}h_2 + a_{13}h_1.$$

$$\begin{aligned} f_n = c^{g_n} &= c^{h_n - n} \\ &= c^{a_{11}h_3 + a_{12}h_2 + a_{13}h_1 - n} \\ &= \left(c^{h_3 - 3}\right)^{a_{11}} \cdot \left(c^{h_2 - 2}\right)^{a_{12}} \cdot \left(c^{h_1 - 1}\right)^{a_{13}} \cdot c^{3a_{11} + 2a_{12} + a_{13} - n} \\ &= f_3^{a_{11}} \cdot f_2^{a_{12}} \cdot f_1^{a_{13}} \cdot c^{3a_{11} + 2a_{12} + a_{13} - n}. \end{aligned}$$

From Fermat's Little Theorem we know that, for a prime p and a not divisible by p,

$$a^n \mod p = a^{n \mod p-1} \mod p.$$

So, compute the matrix modulo $10^9 + 6$ rather than $10^9 + 7$.