数学 02

- 快速幂
- 快速乘
- 矩阵类与矩阵快速幂
- 矩阵加速
- 题目选讲

快速幂

用于快速计算 $a^b \ mod \ m$，时间复杂度为 $O(logb)$

举例来讲，为了计算 3^7，就是计算 $3^{(111)_2} = 3^4 \cdot 3^2 \cdot 3^1$
也就是说，只要知道了 3^4，3^2，3^1 的值，就可以快速计算出答案
事实上，除了最后一个 3^1，每一位都是后面的平方，即 $3^4 = (3^2)^2$，$3^2 = (3^1)^2$，这样就不用傻傻跑 7 次循环了

代码

```
int qpow(int a, int b, int m)
{
    int ans = 1;
    while(b) {
        if(b & 1) ans *= a, ans %= m;
        a *= a, a %= m, b >>= 1;
    }
    return ans;
}
```

快速乘

用于计算 $a \cdot b \ mod \ m$，时间复杂度为 $O(logb)$

为何需要快速乘

细心的同学会发现，计算机中乘法计算是 $O(1)$ 的，快速乘反而更慢了，应该称之为龟速乘才对

既然如此，为何需要快速乘呢?

因为在算法竞赛中，偶尔会遇到整数相乘爆精度问题，例如两个 64 位整数相乘会爆 $long\ long\ int$，所以我们需要牺牲一定的时间复杂度来保证准确性，从而有了快速乘的说法

快速乘求法

与快速幂相似，我们只需要把乘法换成加法即可

算法时间复杂度为 $O(logb)$

```
int qmul(int a, int b, int m)
{
    int ans = 0;
    while(b) {
        if(b & 1) ans += a, ans %= m;
        a += a, a %= m, b >>= 1;
    }
    return ans;
}
```

矩阵类与矩阵快速幂

矩阵快速幂是一种快速计算递推式的算法，但是对代码能力要求较高，需要正确编写矩阵类，然后调用矩阵快速幂函数

矩阵快速幂简单来讲，就是把快速幂的基换成单位矩阵，乘法换成矩阵乘法，其中单位矩阵形如

$$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

模板见 notes.md

矩阵加速

考虑如下问题

$$f_x = a_1 f_{x-1} + a_2 f_{x-2} + a_3 f_{x-3} + \ldots + a_n f_{x-n}$$

给定 t，求 f_t

注意：t 通常很大，线性递推难以计算

解法

构造矩阵递推式：

$$\begin{bmatrix} f_x \\ f_{x-1} \\ f_{x-2} \\ \dots \\ f_{x-n+1} \end{bmatrix} = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_{n-1} & a_n \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix} * \begin{bmatrix} f_{x-1} \\ f_{x-2} \\ f_{x-3} \\ \dots \\ f_{x-n} \end{bmatrix} = \dots = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_{n-1} & a_n \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}^{x-n} * \begin{bmatrix} f_n \\ f_{n-1} \\ f_{n-2} \\ \dots \\ f_1 \end{bmatrix}$$

归纳得到

$$F(x) = A \cdot F(x-1) = A^2 \cdot F(x-2) = \dots = A^{x-n} F(n)$$

其中:

$$F(x) = \begin{bmatrix} f_x \\ f_{x-1} \\ f_{x-2} \\ \dots \\ f_{x-n+1} \end{bmatrix}, F(n) = \begin{bmatrix} f_n \\ f_{n-1} \\ f_{n-2} \\ \dots \\ f_1 \end{bmatrix}, A = \begin{bmatrix} a_1 & a_2 & a_3 & \dots & a_{n-1} & a_n \\ 1 & 0 & 0 & \dots & 0 & 0 \\ 0 & 1 & 0 & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & 1 & 0 \end{bmatrix}$$

利用矩阵快速幂，计算递推式第 t 项，可以在 $O(n^3 logt)$ 时间内求出答案，**记得特判前几项**

653. 快速幂取模

输入三个正整数 x，p，m，计算 $x^p\ mod\ m$

规定 $1 \leq x \leq 10000$，$2 \leq m \leq 10000$，$1 \leq p \leq 10^{15}$

题解

模板题，直接上快速幂

2560. 矩阵，加速！

给定数列 a，$a_1 = a_2 = a_3 = 1$，当 $i > 3$ 时，有 $a_i = a_{i-1} + a_{i-3}$，求 a 的第 n 项对 $10^9 + 7$ 取余的值

规定，$1 \le n \le 2,000,000,000$

题解

构造矩阵如下

$$\begin{bmatrix} a_x \\ a_{x-1} \\ a_{x-2} \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \cdot \begin{bmatrix} a_{x-1} \\ a_{x-2} \\ a_{x-3} \end{bmatrix}$$

接下来用矩阵快速幂递推即可，递推第 x 项的时间复杂度 $O(n^3logx)$，其中 $n = 3$