原创 【Gym 100633J】Ceizenpok’s formula 扩展Lucas详解

$C_n^m \ mod \ p$, p 不为素数

首先对p进行因式分解，$p = p_1^{a_1} p_2^{a_2} p_3^{a_3} \dots p_n^{a_n}$

然后用中国剩余定理合并

$$\begin{cases} C_n^m \ mod \ p_1^{a_1} \\ C_n^m \ mod \ p_2^{a_2} \\ \quad \dots \\ C_n^m \ mod \ p_n^{a_n} \end{cases}$$

现在的问题是怎么求出 $C_n^m \ mod \ p^t$

$$C_n^m \ mod \ p^t = \frac{n!}{m!(n-m)!} \ mod \ p^t$$

因为$m!$、$(n-m)!$ 不一定和 p^t 互质, 所以$m!$、$(n-m)!$ 的逆元不一定存在

因此我们可以化简一下

$\frac{\frac{n!}{p^x}}{\frac{m!}{p^y} \frac{(n-m)!}{p^z}} p^{x-y-z} \ mod \ p^t$,这样形如$\frac{n!}{p^x}$与 p^t一定互质，逆元一定存在

那我们怎么计算$\frac{n!}{p^x} \ mod \ p^t$ ，我们可以先计算$n! \ mod \ p^t$

为了方便理解，我们先假设$n = 22, p = 3, t = 2$

22!

$= (1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7 \cdot 8 \cdot 9 \cdot 10 \cdot 11 \cdot 12 \cdot 13 \cdot 14 \cdot 15 \cdot 16 \cdot 17 \cdot 18 \cdot 19 \cdot 20 \cdot 21) \ mod \ 3^2$

$= (1 \cdot 2 \cdot 4 \cdot 5 \cdot 7 \cdot 8)(10 \cdot 11 \cdot 13 \cdot 14 \cdot 16 \cdot 17)(19 \cdot 20 \cdot 22)(3 \cdot 6 \cdot 9 \cdot 12 \cdot 15 \cdot 18 \cdot 21)$

$= 3^7 \cdot 7!(1 \cdot 2 \cdot 4 \cdot 5 \cdot 7 \cdot 8)^2(19 \cdot 20 \cdot 22) \ mod \ 3^2$

$$n! \ mod \ p^t = p^{\lfloor \frac{n}{p} \rfloor} \left\lfloor \frac{n}{p} \right\rfloor !\left(\sum_{i=1, i \neq 0 \ mod \ p}^{p^t} i\right)^{\lfloor \frac{n}{p^t} \rfloor} \left(\sum_{i=\lfloor \frac{n}{p^t} \rfloor p^t, i \neq 0 \ mod \ p^t}^{n} i\right)$$

因为我们要保证互质，逆元才存在，所以$p^{\lfloor \frac{n}{p} \rfloor}$要被除掉

我们定义$f(n) = \frac{n!}{p^x}$

$$f(n) = f\left(\left\lfloor \frac{n}{p} \right\rfloor\right)\left(\sum_{i=1, i \neq 0 \ mod \ p}^{p^t} i\right)^{\lfloor \frac{n}{p^t} \rfloor} \left(\sum_{i=\lfloor \frac{n}{p^t} \rfloor p^t, i \neq 0 \ mod \ p^t}^{n} i\right)$$

$\frac{\frac{n!}{p^x}}{\frac{m!}{p^y} \frac{(n-m)!}{p^z}} p^{x-y-z} \ mod \ p^t$

$= \frac{f(n)}{f(m)f(n-m)} p^{x-y-z} \ mod \ p^t$ ，那么现在就可以求逆元了

接下来讲解如何求解指数

$$n! \ mod \ p^t = p^{\lfloor \frac{n}{p} \rfloor} \left\lfloor \frac{n}{p} \right\rfloor !\left(\sum_{i=1, i \neq 0 \ mod \ p}^{p^t} i\right)^{\lfloor \frac{n}{p^t} \rfloor} \left(\sum_{i=\lfloor \frac{n}{p^t} \rfloor p^t, i \neq 0 \ mod \ p^t}^{n} i\right)$$

我们令 $g(n) = x$，显然 $p^{\left\lfloor \frac{n}{p} \right\rfloor}$ 是我们需要的指数，但是 $\left\lfloor \frac{n}{p} \right\rfloor !$ 可能还有 P 的倍数

所以, $g(n) = \left\lfloor \frac{n}{p} \right\rfloor + g(\left\lfloor \frac{n}{p} \right\rfloor)$

```
#include<bits/stdc++.h>
using namespace std;
typedef long long ll;
ll power(ll a, ll b, ll n)  // 快速幂
{
        a %= n;
        ll ans = 1;
        while(b) {
                if(b & 1) ans = (ans * a) % n;
                a = (a * a) % n;
                b >>= 1;
        }
        return ans;
}
ll exgcd(ll a, ll b, ll &x, ll &y) // 扩展欧几里得
{
        if(b == 0) {
                x = 1;
                y = 0;
                return a;
        }
        ll d = exgcd(b, a % b, x, y);
        ll t = x;
        x = y;
        y = t - a / b * y;
        return d;
}
ll inverse(ll a, ll p)  // 逆元
{
        ll x, y;
        exgcd(a, p, x, y);
        x = (x % p + p) % p;
        return x;
}
ll CRT(ll *a, ll *m, ll n) // 中国剩余定理
{
        ll M = 1, ans = 0, x, y;
        for(ll i = 1; i <= n; i++) M *= m[i];
        for(ll i = 1; i <= n; i++) {
                ll w = M / m[i];
                exgcd(w, m[i], x, y);
                x = (x % m[i] + m[i]) % m[i];
                ans = (ans + (a[i] % M + M) % M * w % M * x % M) % M;
        }
        return (ans % M + M) % M;
}
ll calc(ll n, ll x, ll p)  // n! mod p
{
        if(n == 0) return 1;
```


展开阅读全文