

3.数学

最大公约数最小公倍数关系

```
1 ab = pq
```

求因数

```
1 // nlnn 求因数
2 vector<vector<int>> fac(K + 1);
3 for (int i = 1; i <= K; i++) {
4     for (int j = i << 1; j <= K; j += i) {
5         fac[j].push_back(i);
6     }
7 }
```

辗转相除法(欧几里得算法)

作用 求最大公约数(gcd)

```
1 int gcd(int a, int b){
2     if(b == 0) return a;
3     return gcd(b, a % b)
4 }
```

扩展欧几里得算法

```
1
```

素数判定

```
1 bool isPrime(int n){
2     for(int i = 2; i * i <= n; i++){
3         if(n % i == 0) return false;
4     }
5     return n != 1;
6 }
```

埃式筛选(求n以内素数个数)

- [1175. 质数排列](#)
- [204. 计数质数](#)

```
1 int prime[MAX_N]
2 bool isPrime[MAX_N + 1];
3
4 int sieve(int n){
5     int p = 0; //素数个数
6     for(int i = 0; i <= n; i++) isPrime[i] = true;
7     isPrime[0] = isPrime[1] = false;
8     for(int i = 2; i <= n; i++){
9         if(isPrime[i]){
10             prime[p++] = i;
11             for(int j = 2 * i; j <= n; j += i){
12                 isPrime[j] = false;
13             }
14         }
15     }
16 }
```

区间筛法(挑战程序设计竞赛p120)

模运算

- $(a + b) \% p = (a \% p + b \% p) \% p$
- $(a - b) \% p = (a \% p - b \% p) \% p$
- $(a * b) \% p = (a \% p * b \% p) \% p$
- 除法不行
- $(a^b) \% p = ((a \% p)^b) \% p$

结合率：

- $((a+b) \% p + c) \% p = (a + (b+c) \% p) \% p$
- $((ab) \% p * c) \% p = (a * (bc) \% p) \% p$

交换率：

- $(a + b) \% p = (b + a) \% p$
- $(a * b) \% p = (b * a) \% p$

分配率：

- $((a+b)\%p * c) \% p = ((a * c) \% p + (b * c) \% p) \% p$

快速幂

- [link](#)
- [力扣pow\(x,n\)讲快速幂](#)

```
1 typedef long long ll;
2 ll mod ;
3 ll qpow(ll a, ll b) {
4     ll res = 1;
```

```

5     ll base = a;
6     while (b != 0) {
7         if (b & 1) res = res * a % mod;
8         base = base * base % mod
9         b >>= 1;
10    }
11    return res;
12 }

```

快速乘

```

1 typedef long long ll;
2 ll mod ;
3 ll qmul(ll a, ll b){
4     ll res = 0;
5     while(b != 0){
6         if(b & 1) res = (res + a) % mod;
7         a = (a + a) % mod;
8         n >>= 1;
9     }
10 }

```

矩阵快速幂

```

1 typedef long long ll;
2 int mod = 1e9 + 7;
3
4 //矩阵乘法
5 vector<vector<ll>> mul(const vector<vector<ll>>& a, const vector<vector<ll>>& b)
6 {
7     vector<vector<ll>> ans(a.size(), vector<ll>(b[0].size(), 0));
8     for (int i = 0; i < a.size(); ++i)
9     {
10         for (int j = 0; j < b[0].size(); ++j)
11         {
12             for (int k = 0; k < b.size(); ++k)
13             {
14                 ans[i][j] += a[i][k] * b[k][j];
15                 ans[i][j] %= mod;
16             }
17         }
18     }
19     return ans;
20 }
21 //矩阵a的n次方
22 vector<vector<ll>> power(vector<vector<ll>> a, int n)
23 {
24     vector<vector<ll>> ans(a.size(), vector<ll>(a.size(), 0));
25     for (int i = 0; i < a.size(); ++i)
26     {
27         ans[i][i] = 1;
28     } //单位阵
29     while (n != 0)
30     {

```

```

31         if (n & 1)
32         {
33             ans = mul(ans, a);
34         }
35         a = mul(a, a);
36         n >>= 1;
37     }
38     return ans;
39 }

```

- 题目：552 1137（感觉好像要根据状态机找出矩阵，然后才能算，之后看看）

排列数

- [链接](#)

组合数

```

1  #include<bits/stdc++.h>
2  using namespace std;
3  #define ll long long
4  const int mod=1e9+7;
5
6  ll power(ll a,ll b){
7      ll res=1;
8      while(b){
9          if(b&1)res=res*a%mod;
10         b>>=1,a=a*a%mod;
11     }
12     return res;
13 }
14 ll inv(ll x){return power(x,mod-2);}
15 ll C(ll a,ll b){
16     ll t=1;
17     for(int i=1;i<=b;i++)t=t*(a-i+1)%mod*inv(i)%mod;
18     return t;
19 }

```

- [链接1](#)
- [链接2](#)

隔板法

- 在n个元素间插入k个板子、把n个小球放到k个盒子中
- $C(?、?)$