数学基础

试除法判定质数

```
bool is_prime(int x){
    if (x < 2) return false;
    for (int i = 2; i <= x / i; i ++ ){
        if (x % i == 0){
            return false;
        }
    }
    return true;
}
```

试除法分解质因数

```
void divide(int x){
    for (int i = 2; i <= x / i; i ++ ){
        if (x % i == 0){
            int s = 0;
            while (x % i == 0) x /= i, s ++ ;
            cout << i << ' ' << s << endl;
        }
    }
    if (x > 1) cout << x << ' ' << 1 << endl;
    cout << endl;
}
```

试除法求所有约数

```
vector<int> get_divisors(int x){
    vector<int> res;
    for (int i = 1; i <= x / i; i ++ ){
        if (x % i == 0){
            res.push_back(i);
            if (i != x / i) res.push_back(x / i);
        }
    }
    sort(res.begin(), res.end());
    return res;
}
```

约数个数&约数之和

若$N = p_1^{c_1} * p_2^{c_2} * \ldots * {p_k}^{c_k}$
约数个数：$(c_1 + 1) * (c_2 + 1) * \ldots * (c_k + 1)$

```
#include <bits/stdc++.h>
using namespace std;
```

```
typedef long long ll;
const int N = 1e5 + 5, MOD = 1e9 + 7;

int main(){
    int n;
    scanf("%d", &n);
    unordered_map<int, int> primes;
    while(n--){
        int x;
        scanf("%d", &x);
        for (int i = 2; i <= x / i; i ++ ){
            while(x % i == 0){
                x /= i;
                primes[i]++;
            }
        }
        if(x > 1) primes[x]++;
    }
    ll res = 1;
    for(auto prime : primes){
        res = res * (prime.second + 1) % MOD;
    }
    cout << res;
    return 0;
}
```

约数之和：$(p_1^0+p_1^1+...+p_1^{c_1})*...*(p_k^0+p_k^1+...+p_k^{c_k})$

```
#include <bits/stdc++.h>
using namespace std;

typedef long long ll;
const int N = 1e5 + 5, MOD = 1e9 + 7;

int main(){
    int n;
    scanf("%d", &n);
    unordered_map<int, int> primes;
    while(n--){
        int x;
        scanf("%d", &x);
        for (int i = 2; i <= x / i; i ++ ){
            while(x % i == 0){
                x /= i;
                primes[i]++;
            }
        }
        if(x > 1) primes[x]++;
    }
    ll res = 1;
    for(auto prime : primes){
        int p = prime.first, s = prime.second;
        ll t = 1;
```

```
        while(s--) t = (t * p + 1) % MOD;
        res = res * t % MOD;
    }
    cout << res;
    return 0;
}
```