山东大学附属中学信息学奥赛资料

数论-1(2023.11-28)

1.质数（素数）

1.1 质数定义

正整数 1 只有一个正整数因子。任意其他的正整数至少有两个正整数因子，因为他一定能被1和本身整除。

只有两个正因子的正整数称为质数，也称为素数。

质数是指大于 1 的正整数，并且除了 1 和它本身不能被其他的正整数整除。如：$2, 3, 5, 7, 11, ...$

大于 1 的正整数除了质数以为的称为合数。

最小的质数是 2。

1 既不是质数也不是合数。

1.2 判断n是质数

结论：如果 n 是一个合数，那么 n 一定有不超过 $\sqrt{n}$ 的质因子。

正整数 n是质数，当且仅当它不能被任何小于等于 $\sqrt{n}$ 的质数整除。

判断 n 是否为质数方法，就是判断 2 到 $\sqrt{n}$ 之间是否有 n 的因数(质因数)。

试除法判断素数：时间 $O(\sqrt{n})$

```
bool check(long long x){
    if(x<=1)return 0;
    if(x==2)return 1;
    for(long long i=2;i*i<=x;i++)
        if(x%i==0)return 0;
    return 1;
}
```

还可以提前求出 2 到 $\sqrt{n}$ 内的素数，然后试除。

1.3 筛选法求素数表

给你一个整数 n，求出 n 以内的质数。

直接用试除法逐个判断 2 到 n 之间的每一个整数，时间 $O(n\sqrt{n})$。

1.3.1 埃拉托斯尼斯筛法（埃氏筛分）：

埃氏筛法的思想：

任意整数 x 的倍数： $2x, 3x, 4x, \ldots$ 一定不是质数。

埃氏筛法的步骤：

划掉 所有 2 的倍数；

剩下的下一个是 3 ，然后再划掉 3 的倍数；

剩下的下一个是 5 ，然后划掉 5 的倍数；

后面依次剩下的是 $7, 11, 13, 17, \ldots$

结论：剩下的没被划掉的都是质数，因为他没被前面的任何一个数筛掉（没有因数）。

算法实现：

考虑质数 x，把 x 的倍数全部做标记筛掉，而且只需要从 $x * x$ 开始 $(x * x, (x+1) * x, \ldots, (n/x) * x)$ 即可，因为小于他的数一定被更小的质数已经筛掉了。而且如果 x 是合数，那么 x 的倍数也已经被更小的质数筛掉了。

```
memset(is_prime,1,sizeof(is_prime));
is_prime[1]=0;
for(int i=2;i<=n;i++)
    if(is_prime[i]){
        prime[++tot]=i;
        for(int j=i*i;j<=n;j+=i)is_prime[j]=0;
    }
```

时间是 $O(nlogn)$。

1.3.2 欧拉筛法（线性筛法）

埃氏筛法的缺点是有的数被筛多次，如 6 就是被 2 和 3 各筛过 一次，如何只被筛一次。

欧拉筛的原理：保证每个合数 x 只被最小的质数筛一次。

```
memset(is_prime,1,sizeof(is_prime));
is_prime[1]=0;
for(int i=2;i<=n;i++){
    if(is_prime[i])prime[++tot]=i;
    for(int j=1;j<=tot&&prime[j]*i<=n;j++){
        is_prime[prime[j]*i]=0;
        if(i%prime[j]==0)break;
    }
}
```

关键语句是： if(i%prime[j]==0)break;

合数=最小质因子$prime[j]$*最大因数 i

线性筛的步骤：

已经得到了质数 $prime[1], prime[2], .., prime[tot]$, 对于当前的 i ，要筛掉的分别是：

$i * prime[j], i * prime[j+1], .., i * prime[tot]$

先筛掉 $i * prime[j]$，然后判断，如果 i 能被 $prime[j]$ 整除，说明prime[j]一定是 $i = prime[j] * \frac{i}{prime[j]}$ 最小质因数，prime[j]数组中的质数是递增的，那么prime[j+1]就不是 i*prime[j+1] 最小质因数（可以由prime[j]筛掉）， 这个合数肯定会被后面的数筛掉（i / prime[j]. prime[j+1]),因此直接break。

如i=15时，已经得到的质数是2,3,5,7,11,13.

筛掉 $15 * 2 = 30, 15 * 3 = 45$, i%prime[j]=0了，后面的 $15 * 5, 15 * 7, 15 * 11, 15 * 13$ 就不用筛了。

因为$i = 15 = 3 * 5$，所以 如 $15 * 5 = 3 * 5 * 5$，就可以由后面当 $i = 25$ 时$*prime[2] = 3$ 筛掉。

同理 $15 * 7 = 3 * 5 * 7$ 由后面的 $i = 35$时 $*prime[2] = 3$ 筛掉。

i	质数	筛掉的数
2	2	4
3	2,3	6,9
4	2,3	8
5	2,3,5	10,15,25
6	2,3,5	12
7	2,3,5,7	14,21,35,49
8	2,3,5,7	16
9	2,3,5,7	18,27
10	2,3,5,7	20
11	2,3,5,7,11	22,33,55,77,121
12	2,3,5,7,11	24
13	2,3,5,7,11	26,39,65,91,143
14	2,3,5,7,11,13	28
15	2,3,5,7,11,13	30,45

$i = 12$时，12×2筛掉了 24 ,break。$12 * 3 = 36$ 不用筛，后面会有 $i = 12/2 * 3 = 18$ 筛掉（$2 * 18 = 36$)。

当i是质数时，筛掉的是i*(i以内的质数)那些数；如i=13.

当i是合数数时，筛掉的是i*(不超过i的最小质因子)：如i=15，35.

因为每个数只被筛一次，所以欧拉筛的时间是 $O(n)$。

P3383 【模板】线性筛素数

2.唯一分解定理

唯一分解定理（基本算数定理）：任意一个大于1的正整数n，都能被唯一表示成素数乘积的形式。

素数$p_1 < p_2 < \ldots < p_k$，$a_1, a_2, \ldots, a_k$是指数。

$n = p_1^{a_1} \times p_2^{a_2} \times .. \times p_k^{a_k}$

性质1：设 $d(n)$ 为 n 的正因子的个数，$d(n) = (a_1+1).(a_2+1)..(a_k+1)$

性质2：设 $\phi(n)$ 为 n 的所有因子的和，$\phi(n) = \frac{p_1^{a_1+1}-1}{p_1-1}.\frac{p_2^{a_2+1}-1}{p_2-1}..\frac{p_k^{a_k+1}-1}{p_k-1}$

性质3：设$a = p_1^{a_1} \times p_2^{a_2} \times .. \times p_k^{a_k}$，$b = p_1^{a_1} \times p_2^{a_2} \times .. \times p_k^{a_k}$，则有：

$$gcd(a,b) = p_1^{\min(a_1,b_1)} \times p_2^{\min(a_2,b_2)} \times .. \times p_k^{\min(a_k,b_k)}$$

$$lcm(a,b) = p_1^{\max(a_1,b_1)} \times p_2^{\max(a_2,b_2)} \times .. \times p_k^{\max(a_k,b_k)}$$

性质4：$n!$ 的质因子分解中，质数 p 的幂为：

$$\frac{n}{p} + \frac{n}{p_2} + \frac{n}{p_3} + \ldots$$

3.质因数分解

结论：如果 n 有大于 $\sqrt{n}$ 的 **质因子**，则只有一个，且幂为1。如果有两个，则乘积大于n了。

3.1 试除法

我们可以只枚举小于等于 $\sqrt{n}$ 的数并判定其是否为 n 的因数，若是则从 n 中除去;

若结束后，剩余于1，那么此时的 剩余那个大于 $\sqrt{n}$ 的质因数。时间复杂度O($\sqrt{n}$)。

如：$n = 12$ 时最后剩余 1 , $n = 42$ 最后的剩余是 7。

```
tot=0;
for(int i=2;i*i<=n;i++)
    if(n%i==0){
        p[++tot]=i;a[tot]=0;
        while(n%i==0)n=n/i,a[tot]++;
    }
if(n>1)p[++tot]=n,a[tot]=1;
```

3.2 筛选法

试除法是枚举 2 到 $\sqrt{n}$ 的每一个数，显然合数是没必要枚举的，如果提前求出他们中的素数，只枚举素数即可了，这样可以节省时间。

第一步：先求出 $\sqrt{n}$ 以内的素数表。

第二步：用素数表试除。注意当$prime[i] * prime[i] > n$ 时结束。

若循环结束后，n 仍大于1，那么此时的 n 就是那个大于 $\sqrt{n}$ 的质因数，并且幂次是 1。

```
void init(int n){
    memset(is_p,1,sizeof(is_p));
    is_p[1]=0;
    for(int i=2;i*i<=n;i++){
        if(is_p[i]==1){
            prime[++m]=i;
            for(int j=i*i;j<=n;j+=i) is_p[j]=0;
        }
```

```
    }
}
void divid(int n){
    int tem=n;
    for(int i=1;i<=m;i++){
        if(prime[i]*prime[i]>tem)break;
        if(n%prime[i]==0){
            p[++tot]=prime[i];a[tot]=0;
            while(n%prime[i]==0)a[tot]++,n/=prime[i];
        }
    }
    if(n>1)p[++tot]=n,a[tot]++;
}
```

3.3 记录最小质因子

先求出n以内的每个数的最小质因子。

```
void init(int n){
    memset(is_p,0,sizeof(is_p));
    for(int i=2;i<=n;i++){
        if(prime[i]==0){
            prime[i]=i;
            for(int j=i+i;j<=n;j+=i) if(!prime[j])prime[j]=i;
        }
    }
}
void divid(int n){
    while(n!=1){
        p[++tot]=prime[n];a[tot]=0;
        while(n%p[tot]==0)a[tot]++,n/=p[tot];
    }
}
```

数论训练题目1（2023.11.28）：

1.P1075质因数分解

题目描述

已知正整数 n 是两个不同的质数的乘积，试求出两者中较大的那个质数。

输入格式

输入一个正整数 n。

输出格式

输出一个正整数 p，即较大的那个质数。

样例 #1

样例输入 #1

```
21
```

样例输出 #1

```
7
```

提示

$1 \le n \le 2 \times 10^9$

2. 【模板】线性筛素数

题目背景

本题已更新，从判断素数改为了查询第 k 小的素数
提示：如果你使用 `cin` 来读入，建议使用 `std::ios::sync_with_stdio(0)` 来加速。

题目描述

如题，给定一个范围 n，有 q 个询问，每次输出第 k 小的素数。

输入格式

第一行包含两个正整数 n, q，分别表示查询的范围和查询的个数。

接下来 q 行每行一个正整数 k，表示查询第 k 小的素数。

输出格式

输出 q 行，每行一个正整数表示答案。

样例 #1

样例输入 #1

```
100 5
1
2
3
4
5
```

样例输出 #1

```
2
3
5
7
11
```

提示

【数据范围】
对于 100% 的数据，$n = 10^8$，$1 \le q \le 10^6$，保证查询的素数不大于 n。

3. CH3101 阶乘分解

描述

给定整数 N(1≤N≤10^6)，试把阶乘 N! 分解质因数，按照算术基本定理的形式输出分解结果中的 p_i 和 c_i 即可。

输入格式

一个整数N。

输出格式

N! 分解质因数后的结果，共若干行，每行一对pi, ci，表示含有pi^ci项。按照pi从小到大的顺序输出。

样例输入

```
5
```

样例输出

```
2 3
3 1
5 1
```

样例解释

5! = 120 = 2^3 * 3 * 5

4.poj2689 Prime Distance

给出一个区间 [l, r] 求其中相邻的距离最近和最远的素数对 。

 其中 1 <= l < r <= 2,147,483,647, r - l <= 1e6 。

输入：

2 17

14 17

输出：

2,3 are closest, 7,11 are most distant.

There are no adjacent primes.