埃氏筛与线性筛

我们已知的质数判断方法，是对每个数进行判断其是否是质数完成的。在优化了判断区间后，时间复杂度O(sqrt(n))。但是如果要找到小于等于N的全部质数，该方法明显太慢了，时间复杂度需要O(n*sqrt(n))。质数筛法可以直接找出小于等于N的全部质数，并且存放在数组中，之后判断某个数是否是质数的时间是O(1)。

埃拉托斯特尼筛法

由希腊数学家埃拉托斯特尼所提出。埃氏筛基于这样一个数学事实：大于1的一个数的**任意倍数**，一定是合数。并且任意合数一定是某一质数的倍数（整数唯一分解定理）。使用埃氏筛筛选质数的过程如下：

1. 设置一个bool类型的is_prime数组，用于记录每个数字是否是质数。**is_prime[i]true表示i是质数，is_prime[i]false表示i不是质数**
2. 用变量i遍历[2,N]之间的全部数
   a. 如果i已经被标为非质数，则跳过该数
   b. 如果i被标为质数，则以i为筛子，将其倍数全部标为非质数

埃筛的实现

```
void o_prime(int n)
{
    // 初始化is_prime数组，假设所有数都是素数
    for (int i = 2; i <= n; i++)
        is_prime[i] = true;

    // 遍历从2到sqrt(n)的每个数i
    for (int i = 2; i * i <= n; i++)
    {
        if (is_prime[i] == true) // 如果i是素数
        {
            // 根据i的值设置步长x
            int x = i == 2 ? 1 : 2;

            // 遍历i的倍数j，将i*j标记为非素数
            for (int j = i; j * j <= n; j += x)
            {
                is_prime[i * j] = false; // 标记i*j为非素数
            }
        }
    }
}
```

1. 初始化is_prime数组，将2到n的所有数标记为true（即假设它们都是素数）。
2. 遍历从2到sqrt(n)的每个数i：
   a. 如果i是素数，则将其倍数标记为非素数（is_prime[i*j] = false）。
   b. 对于偶数和奇数分别进行优化处理：
      i. 当i为2时，步长为1（因为2是最小的素数，且所有偶数都应被标记为非素数）。
      ii. 当i为其他素数时，步长为2（跳过所有偶数，减少不必要的计算）

欧拉筛法

埃氏筛法仍有优化空间，**它会将一个合数重复多次标记**。有没有什么办法省掉无意义的步骤呢？我们有一种新的筛法，**称为欧拉筛或者线性筛法**，能够让每个合数都只被标记一次（被其最小质因子筛选），那么时间复杂度就降到 O(n) 了。使用欧拉筛筛选质数的过程如下：

1. 初始化is_prime数组，将2到n的所有数标记为true（即假设它们都是素数）。
2. 遍历从2到n的每个数i：
   a. 如果i是素数，则将其加入prime数组。
   b. 对于已经找到的每个素数p，将i*p标记为非素数（is_prime[i*p] = false）。
   c. 如果i能被当前素数p整除，则停止进一步处理（因为更大的倍数已经被更小的素数处理过）。

```
void o_prime(int n)
{
    for (int i = 2; i <= n; i++) // 初始化is_prime数组，假设所有数都是素数
        is_prime[i] = true;

    for (int i = 2; i <= n; i++) // 遍历每个数i
    {
        if (is_prime[i] == true) // 如果i是素数
            prime.push_back(i);  // 将其加入prime数组

        // 遍历已经找到的素数p，将i*p标记为非素数
        for (int j = 0; j < prime.size() && i * prime[j] <= n; j++)
        {
            is_prime[i * prime[j]] = false; // 标记i*p为非素数

            // 如果i能被当前素数p整除，则停止进一步处理
            if (i % prime[j] == 0)
                break;
        }
    }
}
```

质数判断方法的时间复杂度对比

<table>
<tr><td></td><td>【暴力枚举】</td><td>【优化枚举】</td><td>【埃氏筛】</td><td>【欧拉筛】</td></tr>
<tr><td>判断一个数是否质数</td><td>O(n)</td><td>O($\sqrt{n}$)</td><td>O(n loglog n)</td><td>O(n)</td></tr>
<tr><td>找出小于等于N的全部质数</td><td>O(n^2)</td><td>O($n * \sqrt{n}$)</td><td colspan="2" rowspan="2">上述是找出小于等于 n 的全部
（即使只判断一个）
之后判断的时间是 O(1)</td></tr>
<tr><td colspan="2">每个数都要判断一次</td></tr>
</table>