

质数筛选/素数因子

若 a 是一个合数，那么一定可以被表示成 $a = pq$ ，其中 $p, q > 1$ ，我们不难推出其中的一个数一定不超过 \sqrt{a} ，因为如果全都超过 \sqrt{a} ， pq 必然大于 a ，这样等式就不成立了，再进一步，一个合数一定存在一个小于等于 \sqrt{a} 的质数 p ，使 $a = p \cdot q$ 成立，推到这里相比各位已经豁然开朗了，我们在遍历的时候将边界设置为 \sqrt{a} 就完全可以找到数 n 除去1与自己的因数，因此给出如下代码：

C++

```
1 bool isprime(int n){
2     for(int i = 2; i*i <= n; i++){
3         if(n % i == 0){
4             return false;
5         }
6     }
7     return true;
8 }
```

我们就将原先的 $O(n)$ 复杂度降到了 $O(\sqrt{a})$

python 版本的快速质数判断：

Python

```
1 def is_prime(n):
2     """Return True if n is a prime number."""
3     if n <= 1:
4         return False
5     if n <= 3:
6         return True
7     if n % 2 == 0 or n % 3 == 0:
8         return False
9     i = 5
10    while i * i <= n:
11        if n % i == 0 or n % (i + 2) == 0:
12            return False
13        i += 6
14    return True
15
16 # 使用示例
17 number = 29
18 if is_prime(number):
19     print(f"{number} is a prime number.")
20 else:
21     print(f"{number} is not a prime number.")
```