

质数判断、质因子分解、质数筛

前置知识

讲解041 - 同余原理

讲解056、057 - 并查集，本节课的题目3需要

判断值较小的数字是否为质数

判断值较大的数字是否为质数，了解 $Miller-Rabin$ 测试的大概过程 + 会用模版即可

某个数字所有质数因子的分解，掌握最常用的方法足够了

找出 $1 \sim n$ 范围内所有的质数，埃氏筛、欧拉筛，其实掌握埃氏筛足够

注意：

讲解097、讲解098、讲解099，可以称为"**不用多问为什么**"专题

有兴趣可以翻帖子看证明，用纸和笔跟着推一遍是最好的方式

因为证明麻烦，并且证明过程没啥扩展性，记住用法和模版即可，当做原子技能使用

质数判断、质因子分解、质数筛

题目1

判断较小的数字是否是质数

一个数字 n ，从 2 开始到 \sqrt{n} ，看看这些数字能否被 n 整除即可，时间复杂度 $O(\sqrt{n})$

质数判断、质因子分解、质数筛

题目2

判断较大的数字是否是质数(Miller-Rabin测试)

测试链接：<https://www.luogu.com.cn/problem/U148828>

判断 n 是否是质数，Miller-Rabin测试大概过程：

- 1，每次选择 $1 \sim n-1$ 范围上的随机数字，或者指定一个比 n 小的质数，进行测试
- 2，测试过程的数学原理不用纠结，不重要，因为该原理除了判断质数以外，不再用于别的方面
- 3，原理：费马小定理、Carmichael(卡米切尔数)、二次探测定理(算法导论31章)、乘法同余、快速幂
- 4，经过 s 次Miller-Rabin测试， s 越大出错几率越低，但是速度也会越慢，一般测试20次以内即可

重点是用法

因为有乘法同余，所以想验证任意的long类型的数字，需要注意位数的事情

代码中都标记好了，我们说明一下：java模版、C++模版

时间复杂度 $O(s * (\log n)^3)$ 的三次方)，速度很快

质数判断、质因子分解、质数筛

题目3

质因子分解的过程 ->

时间复杂度 $O(\sqrt{n})$

有关质因数分解，课上讲的方法足够了

有兴趣的同学可以继续研究

*pollard_rho*启发式方法分解质因数

```
// 打印所有n的质因子
void f(int n) {
    for (int j = 2; j * j <= n; j++) {
        if (n % j == 0) {
            print(j);
            while (n % j == 0) { n /= j; }
        }
    }
    if (n > 1) {
        print(n);
    }
}
```

按公因数计算最大组件大小

给定一个由不同正整数组成的非空数组 *nums*

如果 *nums[i]* 和 *nums[j]* 有一个大于1的公因子，那么这两个数之间有一条无向边

返回 *nums*中最大连通组件的大小

测试链接：<https://leetcode.cn/problems/largest-component-size-by-common-factor/>

质数判断、质因子分解、质数筛

题目4

给定整数 n ，返回 $1 \sim n$ 范围上所有的质数

埃氏筛，时间复杂度 $O(n * \log(\log n))$ ，我们图解一下

欧拉筛，时间复杂度 $O(n)$ ，很精妙，我们图解一下

其实掌握埃氏筛足够了，因为时间复杂度非常接近线性了，而且常数时间很不错

给定整数 n ，返回 小于 n 的质数的数量

测试链接：<https://leetcode.cn/problems/count-primes/>