

# LeetCode第 172 号问题：阶乘后的零

本文首发于公众号「图解面试算法」，是 [图解 LeetCode](#) 系列文章之一。

同步博客：<https://www.algomooc.com>

题目来源于 LeetCode 上第 172 号问题：阶乘后的零。题目难度为 Easy，目前通过率为 38.0%。

## 题目描述

给定一个整数  $n$ ，返回  $n!$  结果尾数中零的数量。

### 示例 1:

输入：3  
输出：0  
解释： $3! = 6$ ，尾数中没有零。

### 示例 2:

输入：5  
输出：1  
解释： $5! = 120$ ，尾数中有 1 个零。

**说明:** 你算法的时间复杂度应为  $O(\log n)$ 。

## 题目解析

题目很好理解，数阶乘后的数字末尾有多少个零。

最简单粗暴的方法就是先乘完再说，然后一个一个数。

事实上，你在使用暴力破解法的过程中就能发现规律：**这 9 个数字中只有 2（它的倍数）与 5（它的倍数）相乘才有 0 出现。**

所以，现在问题就变成了这个阶乘数中能配 **多少对 2 与 5**。

举个复杂点的例子：

$10! = 【 2 * ( 2 * 2 ) * 5 * ( 2 * 3 ) * ( 2 * 2 * 2 ) * ( 2 * 5 ) 】$

在  $10!$  这个阶乘数中可以匹配两对  $2 * 5$ ，所以  $10!$  末尾有 2 个 0。

可以发现，一个数字进行拆分后 2 的个数肯定是大于 5 的个数的，所以能匹配多少对取决于 5 的个数。（好比现在男女比例悬殊，最多能有多少对异性情侣取决于女生的多少）。

那么问题又变成了 **统计阶乘数里有多少个 5 这个因子**。

需要注意的是，像 25，125 这样的不只含有一个 5 的数字的情况需要考虑进去。

比如  $n = 15$ 。那么在  $15!$  中有 3 个 5（来自其中的 5，10，15），所以计算  $n/5$  就可以。

但是比如  $n=25$ ，依旧计算  $n/5$ ，可以得到 5 个 5，分别来自其中的 5，10，15，20，25，但是在 25 中其实是包含 2 个 5 的，这一点需要注意。

所以除了计算  $n/5$  , 还要计算  $n/5/5$  ,  $n/5/5/5$  ,  $n/5/5/5/5$  , ...,  $n/5/5/5,,,/5$  直到商为0, 然后求和即可。

## 代码实现

```
public class Solution {  
    public int trailingZeroes(int n) {  
        return n == 0 ? 0 : n / 5 + trailingZeroes(n / 5);  
    }  
}
```