

70 爬楼梯

题目描述

假设你正在爬楼梯。需要 n 阶你才能到达楼顶。

每次你可以爬 1 或 2 个台阶。你有多少种不同的方法可以爬到楼顶呢？

注意：给定 n 是一个正整数

示例 1：

- 输入：2
- 输出：2
- 解释：有两种方法可以爬到楼顶。
 1. 1 阶 + 1 阶
 2. 2 阶

示例 2：

- 输入：3
- 输出：3
- 解释：有三种方法可以爬到楼顶。
 1. 1 阶 + 1 阶 + 1 阶
 2. 1 阶 + 2 阶
 3. 2 阶 + 1 阶

代码

```
class Solution:
    ...
    k(n) = k(n-1)+k(n-2)
    ...
    def climbStairs(self, n: int) -> int:
        k = list(range(n + 1))
        for i in range(3, n + 1):
            k[i] = k[i - 1] + k[i - 2]
        return k[n]
```

成功 [显示详情](#) >

执行用时 : **52 ms**, 在Climbing Stairs的Python3提交中击败了**79.68%** 的用户

内存消耗 : **12.9 MB**, 在Climbing Stairs的Python3提交中击败了**99.32%** 的用户

进行下一个挑战 :

- 使用最小花费爬楼梯
- 斐波那契数

炫耀一下:     

提交时间	状态	执行用时	内存消耗
几秒前	通过	52 ms	12.9 MB