

0238. 除自身以外数组的乘积

👤 ITCharge ⌚ 大约 1 分钟

- 标签：数组、前缀和
- 难度：中等

题目链接

- [0238. 除自身以外数组的乘积 - 力扣](#)

题目大意

描述： 给定一个数组 `nums`。

要求： 返回数组 `answer`，其中 `answer[i]` 等于 `nums` 中除 `nums[i]` 之外其余各元素的乘积。

说明：

- 题目数据保证数组 `nums` 之中任意元素的全部前缀元素和后缀的乘积都在 32 位整数范围内。
- 请不要使用除法，且在 $O(n)$ 时间复杂度内解决问题。
- **进阶：** 在 $O(1)$ 的额外空间复杂度内完成这个题目。
- $2 \leq \text{nums.length} \leq 10^5$ 。
- $-30 \leq \text{nums}[i] \leq 30$ 。

示例：

- 示例 1:

```
输入: nums = [1,2,3,4]
输出: [24,12,8,6]
```

py

- 示例 2:

```
输入: nums = [-1,1,0,-3,3]
```

py

输出: [0,0,9,0,0]

解题思路

思路 1: 两次遍历

1. 构造一个答案数组 *res*, 长度和数组 *nums* 长度一致。
2. 先从左到右遍历一遍 *nums* 数组, 将 *nums[i]* 左侧的元素乘积累积起来, 存储到 *res* 数组中。
3. 再从右到左遍历一遍, 将 *nums[i]* 右侧的元素乘积累积起来, 再乘以原本 *res[i]* 的值, 即为 *nums* 中除了 *nums[i]* 之外的其他所有元素乘积。

思路 1: 代码

```
class Solution:
    def productExceptSelf(self, nums: List[int]) -> List[int]:
        size = len(nums)
        res = [1 for _ in range(size)]

        left = 1
        for i in range(size):
            res[i] *= left
            left *= nums[i]

        right = 1
        for i in range(size-1, -1, -1):
            res[i] *= right
            right *= nums[i]
        return res
```

py

思路 1: 复杂度分析

- 时间复杂度: $O(n)$ 。
- 空间复杂度: $O(1)$ 。

