238. Product of Array Except Self

Solved ⊘

Medium   🏷 Topics   🔒 Companies

Given an integer array `nums`, return *an array* `answer` *such that* `answer[i]` *is equal to the product of all the elements of* `nums` *except* `nums[i]`.

The product of any prefix or suffix of `nums` is **guaranteed** to fit in a **32-bit** integer.

You must write an algorithm that runs in `O(n)` time and without using the division operation.

Example 1:

```
Input: nums = [1,2,3,4]
Output: [24,12,8,6]
```

Example 2:

```
Input: nums = [-1,1,0,-3,3]
Output: [0,0,9,0,0]
```

```swift
class Solution {
    func productExceptSelf(_ nums: [Int]) -> [Int] {

        var prefixmul = Array(repeating:1, count:nums.count)
        var suffixmul = Array(repeating:1, count:nums.count)

        for i in (1..<nums.count) {
            prefixmul[i] = prefixmul[i-1] * nums[i-1]
        }
        for i in (0..<nums.count-1).reversed() {
            suffixmul[i] = suffixmul[i+1] * nums[i+1]
        }

        for i in (0..<prefixmul.count) {
            suffixmul[i] = suffixmul[i] * prefixmul[i]
        }

        return suffixmul
    }
}
```