238. Product of Array Except Self

Medium

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.

Constraints:

2 <= nums.length <= 105
-30 <= nums[i] <= 30
The product of any prefix or suffix of nums is guaranteed to fit in a 32-bit integer.

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]

Example 1:

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

```
class Solution:
    def productExceptSelf(self, nums: List[int]) -> List[int]:
        answer = [1] * len(nums)

        prefix = 1
        for i in range(len(nums)):
            answer[i] *= prefix
            prefix *= nums[i]

        postfix = 1
        for i in range(len(nums) - 1, -1, -1):
            answer[i] *= postfix
            postfix *= nums[i]

        return answer
```

answer = [1, 1, 1, 1]

i	answer	prefix
0	[1,1,1,1]	1-> **1**
1	[1,1,1,1]	1-> **2**
2	[1,1,2,1]	2-> **6**
3	[1,1,2,6]	6-> **24**

i	answer	postfix
0	[1, 1,2,6]	1-> **4**
1	[1, 1,8,6]	4-> **12**
2	[1,12,8,6]	12-> **24**
3	[24,12,8,6]	24-> **24**

answer = [24,12,8,6]