[Click on the logo to find **Problem Statement**]

Intuition

This code rearranges an array of integers, placing positive integers at even indices and negative integers at odd indices. It iterates through the input array, assigning positive integers to even indices and negative integers to odd indices in the output array. It uses two indices (`pos_index` for positive integers and `neg_index` for negative integers) to keep track of where to place the next element. This approach ensures that positive and negative integers are alternated in the output array, maintaining the relative order of elements.

Approach

1. Initialize an output array `ans` of the same length as the input array `nums` , filled with zeros.
2. Initialize two indices `pos_index` and `neg_index` to 0 and 1 respectively, to track the positions where positive and negative integers will be placed in the output array.
3. Iterate through each element in the input array `nums` :
    - If the element is positive, place it at index `pos_index` in the output array `ans` , then increment `pos_index` by 2 to ensure the next positive integer is placed at the next even index.
    - If the element is negative, place it at index `neg_index` in the output array `ans` , then increment `neg_index` by 2 to ensure the next negative integer is placed at the next odd index.
4. Return the rearranged array `ans` .

Complexity

- Time complexity: O(n)
- Space complexity: O(n)

Code

```python
class Solution:
    def rearrangeArray(self, nums: List[int]) -> List[int]:
        n = len(nums)
        ans = [0] * n
        pos_index, neg_index = 0, 1
        for i in range(n):
            if nums[i] > 0:
                ans[pos_index] = nums[i]
                pos_index += 2
            else:
                ans[neg_index] = nums[i]
                neg_index += 2
        return ans
```

If you want to see more solutions to coding problems, you can visit: