2.Medium\07.Rearange_elements_by_sign.cpp

```cpp
/*
QUESTION:-
You are given a 0-indexed integer array nums of even length consisting of an equal number of positive and negative integers.
You should rearrange the elements of nums such that the modified array follows the given conditions:
Every consecutive pair of integers have opposite signs.
For all integers with the same sign, the order in which they were present in nums is preserved.
The rearranged array begins with a positive integer.
Return the modified array after rearranging the elements to satisfy the aforementioned conditions.


Example 1:

Input: nums = [3,1,-2,-5,2,-4]
Output: [3,-2,1,-5,2,-4]
Explanation:
The positive integers in nums are [3,1,2]. The negative integers are [-2,-5,-4].
The only possible way to rearrange them such that they satisfy all conditions is [3,-2,1,-5,2,-4].
Other ways such as [1,-2,2,-5,3,-4], [3,1,2,-2,-5,-4], [-2,3,-5,1,-4,2] are incorrect because they do not satisfy one or more conditions.
*/

/*
APPROACH:-
Initialize two pointers, pos_ptr and neg_ptr. pos_ptr will point to the first positive integer in the array, and neg_ptr will point to the first negative integer in the array.
Iterate over the array.
If the current integer is positive, swap it with the element at neg_ptr.
Increment pos_ptr by 1.
Increment neg_ptr by 1.
Repeat steps 3-5 until the end of the array is reached.
The array will now be rearranged such that every consecutive pair of integers have opposite signs.
*/

// CODE:-
vector<int> rearrangeArray(vector<int> &nums)
{
    int i = 0; // for +ve integers
    int j = 1; // for -ve integers
    vector<int> ans(nums.size());
    for (int k = 0; k < nums.size(); k++)
    {
        if (nums[k] >= 0)
        {
            ans[i] = nums[k];
            i += 2;
        }
        else
        {
            ans[j] = nums[k];
            j += 2;
        }
    }
    return ans;
}

// TIME COMPLEXITY = O(N)
// SPACE COMPLEXITY = O(0)
```