1.Easy\07.Move_0's_to_end.cpp

```cpp
/*
QUESTION:-
Given an integer array nums, move all 0's to the end of it while maintaining the relative order of the non-zero elements.

Note that you must do this in-place without making a copy of the array.

Example 1:

Input: nums = [0,1,0,3,12]
Output: [1,3,12,0,0]
Example 2:

Input: nums = [0]
Output: [0]
*/

/*
APPROACH:-
-> The idea is while traversing the array if we found any zero then we have to swap it with next non-zero
*/

// CODE:-
// function to find the next non-zero element
int next_nonzero(vector<int> &a, int &j)
{
    while (j < a.size())
    {
        if (a[j] != 0)
            return j;
        j++;
    }
    return -1;
}
void moveZeroes(vector<int> &nums)
{
    int j = -1; // is to find the next non zero element
    // i signifies that upto here all elements are non-zero
    for (int i = 0; i < nums.size(); i++)
    {
        if (nums[i] != 0)
            continue;
        if (j == -1)
            j = i + 1;
        int nxt_non0 = next_nonzero(nums, j);
        if (nxt_non0 == -1)
            return;
        swap(nums[i], nums[nxt_non0]);
    }
}

```