**Move all zeroes to end of array: -**

**Easy Accuracy: 45.51% Submissions: 147K+ Points: 2**

Given an array **arr[]** of **n** positive integers. Push all the zeros of the given array to the right end of the array while maintaining the order of non-zero elements. Do the mentioned change in the array in-place.

**Example 1:**

**Input:**

N = 5

Arr[] = {3, 5, 0, 0, 4}

**Output:** 3 5 4 0 0

**Explanation:** The non-zero elements

preserve their order while the 0

elements are moved to the right.

**Example 2:**

**Input:**

N = 4

Arr[] = {0, 0, 0, 4}

**Output:** 4 0 0 0

**Explanation:** 4 is the only non-zero

element and it gets moved to the left.

**Your Task:**  
You don't need to read input or print anything. Complete the function **pushZerosToEnd()** which takes the array **arr[]**and its size **n** as input parameters and modifies arr[] in-place such that all the zeroes are moved to the right.

**Expected Time Complexity:** O(n)  
**Expected Auxiliary Space:** O(1)

**Constraints:**  
1 ≤ N ≤ 105  
0 ≤ arri ≤ 105

**Code: -**

//{ Driver Code Starts

#include <bits/stdc++.h>

using namespace std;

// } Driver Code Ends

//User function template for C++

class Solution{

public:

void pushZerosToEnd(int arr[], int n) {

// code here

int l = 0;

for(int i=0; i<n; ++i){

if(arr[i] != 0){

swap(arr[i], arr[l]);

++l;

}

}

}

};

//{ Driver Code Starts.

int main() {

int t;

cin >> t;

while (t--) {

int n, i;

cin >> n;

int arr[n];

for (i = 0; i < n; i++) {

cin >> arr[i];

}

Solution ob;

ob.pushZerosToEnd(arr, n);

for (i = 0; i < n; i++) {

cout << arr[i] << " ";

}

cout << "\n";

}

return 0;

}

// } Driver Code Ends

**T.C: - O(N)**

**S.C: - O(1)**