GeeksforGeeks
A computer science portal for geeks
Practice IDE Q&A GeeksQuiz

Move all zeroes to end of array

Given an array of random numbers, Push all the zero's of a given array to the end of the array. For example, if the given arrays is {1, 9, 8, 4, 0, 0, 2, 7, 0, 6, 0}, it should be changed to {1, 9, 8, 4, 2, 7, 6, 0, 0, 0, 0}. The order of all other elements should be same. Expected time complexity is O(n) and extra space is O(1).

There can be many ways to solve this problem. Following is a simple and interesting way to solve this problem.
Traverse the given array 'arr' from left to right. While traversing, maintain count of non-zero elements in array. Let the count be 'count'. For every non-zero element arr[i], put the element at 'arr[count]' and increment 'count'. After complete traversal, all non-zero elements have already been shifted to front end and 'count' is set as index of first 0. Now all we need to do is that run a loop which makes all elements zero from 'count' till end of the array.

Below is C++ implementation of the above approach.

C

```
// A C++ program to move all zeroes at the end of array
#include <iostream>
using namespace std;

// Function which pushes all zeros to end of an array.
void pushZerosToEnd(int arr[], int n)
{
    int count = 0;  // Count of non-zero elements

    // Traverse the array. If element encountered is non-
    // zero, then replace the element at index 'count'
    // with this element
    for (int i = 0; i < n; i++)
        if (arr[i] != 0)
            arr[count++] = arr[i]; // here count is
                                   // incremented

    // Now all non-zero elements have been shifted to
    // front and  'count' is set as index of first 0.
    // Make all elements 0 from count to end.
    while (count < n)
```

```
        arr[count++] = 0;
}

// Driver program to test above function
int main()
{
    int arr[] = {1, 9, 8, 4, 0, 0, 2, 7, 0, 6, 0, 9};
    int n = sizeof(arr) / sizeof(arr[0]);
    pushZerosToEnd(arr, n);
    cout << "Array after pushing all zeros to end of array :\n";
    for (int i = 0; i < n; i++)
        cout << arr[i] << " ";
    return 0;
}
```

Java

```
/* Java program to push zeroes to back of array */
import java.io.*;

class PushZero
{
    // Function which pushes all zeros to end of an array.
    static void pushZerosToEnd(int arr[], int n)
    {
        int count = 0;  // Count of non-zero elements

        // Traverse the array. If element encountered is
        // non-zero, then replace the element at index 'count'
        // with this element
        for (int i = 0; i < n; i++)
            if (arr[i] != 0)
                arr[count++] = arr[i]; // here count is
                                       // incremented

        // Now all non-zero elements have been shifted to
        // front and 'count' is set as index of first 0.
        // Make all elements 0 from count to end.
        while (count < n)
            arr[count++] = 0;
    }

    /*Driver function to check for above functions*/
    public static void main (String[] args)
    {
        int arr[] = {1, 9, 8, 4, 0, 0, 2, 7, 0, 6, 0, 9};
        int n = arr.length;
        pushZerosToEnd(arr, n);
        System.out.println("Array after pushing zeros to the back: ");
        for (int i=0; i<n; i++)
            System.out.print(arr[i]+" ");
    }
}
/* This code is contributed by Devesh Agrawal */
```

Output:

```
Array after pushing all zeros to end of array :
1 9 8 4 2 7 6 9 0 0 0 0
```

Time Complexity: O(n) where n is number of elements in input array.

Auxiliary Space: O(1)

This article is contributed by **Chandra Prakash**. Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above.

104 Comments Category: Arrays

Related Posts:

- Longest Span with same Sum in two Binary arrays
- Count Inversions of size three in a give array
- Find the subarray with least average
- Count triplets with sum smaller than a given value
- Find zeroes to be flipped so that number of consecutive 1’s is maximized
- Reorder an array according to given indexes
- Find maximum value of Sum(i*arr[i]) with only rotations on given array allowed

- Find maximum average subarray of k length

(Login to Rate and Mark)

2 Average Difficulty : **2/5.0**
Based on **7** vote(s)

□ Add to TODO List

□ Mark as DONE

Like Share 51 people like this. Be the first of your friends.

Writing code in comment? Please use code.geeksforgeeks.org, generate link and share the link here.

Contact Us! About Us! Advertise with us!