Find the number of zeroes

Given an array of 1s and 0s which has all 1s first followed by all 0s. Find the number of 0s. Count the number of zeroes in the given array.

Examples:

```
Input: arr[] = {1, 1, 1, 1, 0, 0}
Output: 2

Input: arr[] = {1, 0, 0, 0, 0}
Output: 4

Input: arr[] = {0, 0, 0}
Output: 3

Input: arr[] = {1, 1, 1, 1}
Output: 0
```

We strongly recommend to minimize the browser and try this yourself in time complexity better than O(n).

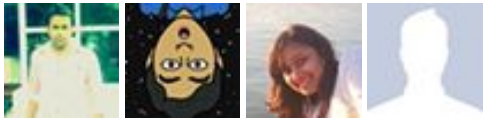
A **simple solution** is to traverse the input array. As soon as we find a 0, we return n – index of first 0. Here n is number of elements in input array. Time complexity of this solution would be O(n).

Since the input array is sorted, we can use **Binary Search** to find the first occurrence of 0. Once we have index of first element, we can return count as n – index of first zero.

Google™ Custom Search

GeeksforGeeks

53,519 people like GeeksforGeeks.



open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

```
// A divide and conquer solution to find count of zeroes in an array
// where all 1s are present before all 0s
#include <stdio.h>

/* if 0 is present in arr[] then returns the index of FIRST occurrence
   of 0 in arr[low..high], otherwise returns -1 */
int firstZero(int arr[], int low, int high)
{
    if (high >= low)
    {
        // Check if mid element is first 0
        int mid = low + (high - low)/2;
        if (( mid == 0 || arr[mid-1] == 1) && arr[mid] == 0)
            return mid;

        if (arr[mid] == 1)  // If mid element is not 0
            return firstZero(arr, (mid + 1), high);
        else  // If mid element is 0, but not first 0
            return firstZero(arr, low, (mid -1));
    }
    return -1;
}

// A wrapper over recursive function firstZero()
int countOnes(int arr[], int n)
{
    // Find index of first zero in given array
    int first = firstZero(arr, 0, n-1);

    // If 0 is not present at all, return 0
    if (first == -1)
        return 0;

    return (n - first);
}

/* Driver program to check above functions */
int main()
{
    int arr[] =   {1, 1, 1, 0, 0, 0, 0, 0};
    int n = sizeof(arr)/sizeof(arr[0]);
    printf("Count of zeroes is %d", countOnes(arr, n));
    return 0;
}
```

Output:

Geometric Algorithms

Popular Posts

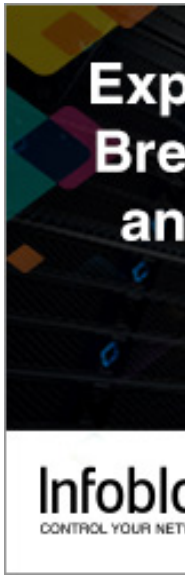
All permutations of a given string
Memory Layout of C Programs
Understanding “extern” keyword in C
Median of two sorted arrays
Tree traversal without recursion and without stack!
Structure Member Alignment, Padding and Data Packing
Intersection point of two Linked Lists
Lowest Common Ancestor in a BST.
Check if a binary tree is BST or not
Sorted Linked List to Balanced BST

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

```
Count of zeroes is 5
```

Time Complexity: O(Logn) where n is number of elements in arr[].

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above



Related Tpoics:

- Remove minimum elements from either side such that 2*min becomes more than max
- Divide and Conquer | Set 6 (Search in a Row-wise and Column-wise Sorted 2D Array)
- Bucket Sort
- Kth smallest element in a row-wise and column-wise sorted 2D array | Set 1
- Find if there is a subarray with 0 sum
- Divide and Conquer | Set 5 (Strassen's Matrix Multiplication)
- Count all possible groups of size 2 or 3 that have sum as multiple of 3
- Sort n numbers in range from 0 to n^2 – 1 in linear time

14 Tweet 3 1

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

Writing code in comment? Please use **ideone.com** and share the link here.

18 Comments GeeksforGeeks

Sort by Newest

Join the discussion…

arjomanD • 15 hours ago

As simple as locating the middle of the array :D

• Reply • Share ›

karthi • 25 days ago

```
import java.io.*;

class NumberOfZeros{

public static int getNumberOfZeros(int[] a){

if(a.length == 0){

return 0;

}

int count = 0;

for(int i=a.length-1;i!=-1&&a[i]!=1;i--){

count++;

}
```

705

Subscribe

Recent Comments

kzs please provide solution for the problem...

Backtracking | Set 2 (Rat in a Maze) · 3 minutes ago

Sanjay Agarwal bool tree::Root_to_leaf_path_given_sum(tree...

Root to leaf path sum equal to a given number · 28 minutes ago

GOPI GOPINATH @admin Highlight this sentence "We can easily...

Count trailing zeroes in factorial of a number · 30 minutes ago

newCoder3006 If the array contains negative numbers also. We...

Find subarray with given sum · 54 minutes ago

newCoder3006 Code without using while loop. We can do it...

Find subarray with given sum · 1 hour ago

Sanjay Agarwal You can also use the this method:...

Count trailing zeroes in factorial of a number · 1 hour ago

AdChoices

▶ Java Source Code

▶ Java Array

▶ Zeroes

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

see more

• Reply • Share ›

Chandu Mannam • 25 days ago

```
#include<stdio.h>

int getposition(int,int,int*);

int main()

{

int a[ ]={1,1,1,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0};

int n,k;

n = sizeof(a)/sizeof(a[0]);

k = getposition(0,n,a);

printf("no of zeros=%d",n-k);

return 0;

}
```

see more

• Reply • Share ›

napender singh • a month ago

```
public class FindZero{

public static void main(String []args){
```

AdChoices
▶ Code Number
▶ Memory Array
▶ Numbers Number
AdChoices
▶ Numbers Number
▶ Number of First
▶ An Array

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

```
int one = findSumOfArr(arr);

int two = findLengthOfArr(arr);

int zero = two - one;

System.out.println(zero);

}

public static int findLengthOfArr(int[] a){

int j = 0;

for(int i=0;i<a.length;i++) j++;="" return="" j;="" }="" public="" static="" int="" fin
temp="0;" for(int="" i="0;i&amp;lt;a.length;i++){" temp="temp" +="" a[i];="" }="" retur
```

:/ • a month ago

In an interview I was asked this question with the additional constraint that the
(Determining N, size, will take usually O(N)) Any ideas on how to approach thi

GeeksforGeeks Mod :/ • a month ago

You can use the idea discussed in the below post.

http://www.geeksforgeeks.org/f...

laksbv :/ • a month ago

http://stackoverflow.com/quest...

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

coder • a month ago

i think this code will not work for an array which contains only zero. Need to ad

```
if(arr[low] == 0){
return low;
}
```

2 ▲ | ▼ • Reply • Share ›

Kartik ➚ coder • a month ago

Please take a closer look. It works for all 0s also. Sample run http://ide

▲ | ▼ • Reply • Share ›

Amit • a month ago

```
#include<stdio.h>

int main()

{

int findzeros(int arr[],int n);

int arr[9]={1,0,0,1,0,0,0,0,1};

printf("Number of zeros: %d",findzeros(arr,9));

return 0;

}

int findzeros(int arr[],int n)

{

if(n>-1)
```

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

Reply • Share ›

Siva Krishna • a month ago

Another possible way is ..first you check in the following way
1 2 4 8 16 32 ...2^k
stop when arr[i] == 0 and then do binary search on sub array i/2 to i for finding
case complexity of O(n) and i think average case complexity is less.

Reply • Share ›

Kartik ↗ Siva Krishna • a month ago

Siva Krishna, Thanks for suggesting this. This solution can be combine
O(Logn). See http://www.geeksforgeeks.org/f...
It can in fact be a good solution if we don't know size of array.

Reply • Share ›

zzer ↗ Siva Krishna • a month ago

it may overflow the boundary of the input array

Reply • Share ›

Siva Krishna ↗ zzer • a month ago

that's an idea..not the exact implementation. You have to do so
implementing that.

Reply • Share ›

zzer ↗ Siva Krishna • a month ago

well, it's really a good idea, this idea has been used in s
of them?

Reply • Share ›

Siva Krishna ↗ zzer • a month ago

1.Find the point where a monotonically increasing functi

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com

2. Find the starting element in an rotated sorted array.

  • Reply • Share ›

Ânon • a month ago

For thé simple solution, you return n - index + 1

  • Reply • Share ›

AA ➦ Ânon • a month ago

SADSADSAD

  • Reply • Share ›

 Subscribe Add Disqus to your site

@geeksforgeeks, **Some rights reserved** **Contact Us!** Powered by **WordPress** & **MooTools**, customized by geeksforgeeks team

open in browser PRO version Are you a developer? Try out the HTML to PDF API pdfcrowd.com