Dashboard / My courses / CS23331-DAA-2023-CSE / Divide and Conquer / 1-Number of Zeros in a Given Array

| | |
|---|---|
| **Started on** | Tuesday, 3 September 2024, 2:22 PM |
| **State** | Finished |
| **Completed on** | Tuesday, 3 September 2024, 2:46 PM |
| **Time taken** | 23 mins 38 secs |
| **Marks** | 1.00/1.00 |
| **Grade** | **10.00** out of 10.00 (**100**%) |

Question **1**

Correct

Mark 1.00 out of 1.00

**Problem Statement**

Given an array of 1s and 0s this has all 1s first followed by all 0s. Aim is to find the number of 0s. Write a program using Divide and Conquer to Count the number of zeroes in the given array.

Input Format

First Line Contains Integer m – Size of array

Next m lines Contains m numbers – Elements of an array

Output Format

First Line Contains Integer – Number of zeroes present in the given array.

**Answer:** (penalty regime: 0 %)

```
#include <stdio.h>
void swap(int* a, int* b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}
int partition(int arr[], int low, int high) {
    int p = arr[low];
    int i = low;
    int j = high;

    while (i < j) {
        while (arr[i] <= p && i <= high - 1) {
            i++;
        }
        while (arr[j] > p && j >= low + 1) {
            j--;
        }
        if (i < j) {
            swap(&arr[i], &arr[j]);
        }
    }
    swap(&arr[low], &arr[j]);
    return j;
}

void quickSort(int arr[], int low, int high) {
    if (low < high) {
        int pi = partition(arr, low, high);
        quickSort(arr, low, pi - 1);
        quickSort(arr, pi + 1, high);
    }
}

int main() {
    int n;
    scanf("%d",&n);
    int arr[n];
    for(int i=0;i<n;i++){
        scanf("%d",&arr[i]);
    }
    int c=0;
    quickSort(arr, 0, n - 1);
    for(int iii=0;iii<n;iii++){
        if(arr[iii]==0){
            c++;
        }
    }
    printf("%d",c);
    return 0;
}

```

| | Input | Expected | Got | |
|---|---|---|---|---|
| ✓ | 5<br>1<br>1<br>1<br>0<br>0 | 2 | 2 | ✓ |
| ✓ | 10<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1 | 0 | 0 | ✓ |
| ✓ | 8<br>0<br>0<br>0<br>0<br>0<br>0<br>0<br>0 | 8 | 8 | ✓ |
| ✓ | 17<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>1<br>0<br>0 | 2 | 2 | ✓ |

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

◄ 5-G-Product of Array elements-Minimum

Jump to...

2-Majority Element ►