



KUMARA GURU S 2024-CSE ▾

K2

**Started on** Wednesday, 17 September 2025, 10:17 AM**State** Finished**Completed on** Wednesday, 17 September 2025, 10:48 AM**Time taken** 31 mins 17 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 %)

```

1 #include <stdio.h>
2 int main() {
3     int m;
4     int c=0;
5     scanf("%d", &m);
6     for (int i = 0; i < m; i++) {
7         int a;
8         scanf("%d", &a);
9         if(a==0)
10             c++;
11     }
12     printf("%d", c);
13     return 0;
14 }
15

```

	Input	Expected	Got	
✓	5 1 1 1 0 0	2	2	✓
✓	10 1 1 1 1 1 1 1 1 1	0	0	✓

	Input	Expected	Got	
✓	8 0 0 0 0 0 0 0 0	8	8	✓
✓	17 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0	2	2	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

KUMARA GURU S 2024-CSE ▾**K2****Started on** Wednesday, 17 September 2025, 10:48 AM**State** Finished**Completed on** Wednesday, 17 September 2025, 11:13 AM**Time taken** 24 mins 22 secs**Marks** 1.00/1.00**Grade** **10.00** out of 10.00 (**100%**)

**Question 1** | Correct Mark 1.00 out of 1.00

Given an array `nums` of size `n`, return *the majority element*.

The majority element is the element that appears more than  $\lfloor n / 2 \rfloor$  times. You may assume that the majority element always exists in the array.

**Example 1:**

**Input:** `nums = [3,2,3]`  
**Output:** 3

**Example 2:**

**Input:** `nums = [2,2,1,1,1,2,2]`  
**Output:** 2

**Constraints:**

- `n == nums.length`
- `1 <= n <= 5 * 10^4`
- `-2^31 <= nums[i] <= 2^31 - 1`

**For example:**

Input	Result
3	3
3 2 3	
7	2
2 2 1 1 1 2 2	

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

```

1 #include <stdio.h>
2 #include <stdlib.h>
3
4 int majorityElement(int nums[], int numsSize) {
5     int candidate = 0;
6     int count = 0;
7
8     for (int i = 0; i < numsSize; i++) {
9         if (count == 0) {
10             candidate = nums[i];
11             count = 1;
12         } else if (candidate == nums[i]) {
13             count++;
14         } else {
15             count--;
16         }
17     }
18
19     return candidate;
20 }
21
22 int main() {
23     int n;
24     scanf("%d", &n);
25     int nums[n];
26     for (int i = 0; i < n; i++) {
27         scanf("%d", &nums[i]);
28     }
29
30     int result = majorityElement(nums, n);
31     printf("%d\n", result);
32
33     return 0;
34 }
```

	Input	Expected	Got	
✓	3 3 2 3	3	3	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

KUMARA GURU S 2024-CSE ▾**K2****Started on** Wednesday, 17 September 2025, 11:15 AM**State** Finished**Completed on** Wednesday, 17 September 2025, 11:31 AM**Time taken** 16 mins 24 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 a sorted array and a value x, the floor of x is the largest element in array smaller than or equal to x. Write divide and conquer algorithm to find floor of x.

**Input Format**

First Line Contains Integer n – Size of array  
 Next n lines Contains n numbers – Elements of an array  
 Last Line Contains Integer x – Value for x

**Output Format**

First Line Contains Integer – Floor value for x

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

```

1 #include <stdio.h>
2 int findFloor(int arr[], int low, int high, int x) {
3     if (low > high) {
4         return -1;
5     }
6
7     if (x < arr[low]) {
8         return -1;
9     }
10    int mid = low + (high - low) / 2;
11    if (arr[mid] == x || (arr[mid] < x && (mid + 1 > high || arr[mid + 1] > x))) {
12        return arr[mid];
13    }
14    else if (arr[mid] > x) {
15        return findFloor(arr, low, mid - 1, x);
16    }
17    else {
18        return findFloor(arr, mid + 1, high, x);
19    }
20 }
21
22 int main() {
23     int n, x;
24
25     scanf("%d", &n);
26
27     int arr[n];
28     for (int i = 0; i < n; i++) {
29         scanf("%d", &arr[i]);
30     }
31     scanf("%d", &x);
32     int floorValue = findFloor(arr, 0, n - 1, x);
33
34     if (floorValue == -1) {
35     }
36     else {
37         printf("%d\n", floorValue);
38     }
39
40     return 0;
41 }
```

	<b>Input</b>	<b>Expected</b>	<b>Got</b>	
✓	6 1 2 8 10 12 19 5	2	2	✓
✓	5 10 22 85 108 129 100	85	85	✓
✓	7 3 5 7 9 11 13 15 10	9	9	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

KUMARA GURU S 2024-CSE ▾**K2****Started on** Wednesday, 17 September 2025, 11:14 AM**State** Finished**Completed on** Wednesday, 17 September 2025, 11:24 AM**Time taken** 10 mins 25 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 a sorted array of integers say arr[] and a number x. Write a recursive program using divide and conquer strategy to check if there exist two elements in the array whose sum = x. If there exist such two elements then return the numbers, otherwise print as "No".

Note: Write a Divide and Conquer Solution

**Input Format**

First Line Contains Integer n – Size of array

Next n lines Contains n numbers – Elements of an array

Last Line Contains Integer x – Sum Value

**Output Format**

First Line Contains Integer – Element1

Second Line Contains Integer – Element2 (Element 1 and Elements 2 together sums to value "x")

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

```

1 #include <stdio.h>
2 void findSumPair(int arr[], int left, int right, int x) {
3     if (left >= right) {
4         printf("No\n");
5         return;
6     }
7
8     int currentSum = arr[left] + arr[right];
9
10    if (currentSum == x) {
11        printf("%d\n", arr[left]);
12        printf("%d\n", arr[right]);
13        return;
14    } else if (currentSum < x) {
15        findSumPair(arr, left + 1, right, x);
16    } else {
17        findSumPair(arr, left, right - 1, x);
18    }
19}
20 int main() {
21     int n, x;
22     scanf("%d", &n);
23     int arr[n];
24     for (int i = 0; i < n; i++) {
25         scanf("%d", &arr[i]);
26     }
27     scanf("%d", &x);
28     findSumPair(arr, 0, n - 1, x);
29
30     return 0;
31 }
```

	Input	Expected	Got	
✓	4 2 4 8 10 14	4 10	4 10	✓
✓	5 2 4 6 8 10 100	No	No	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

KUMARA GURU S 2024-CSE ▾

K2

**Started on** Wednesday, 17 September 2025, 11:13 AM**State** Finished**Completed on** Wednesday, 17 September 2025, 11:19 AM**Time taken** 5 mins 59 secs**Marks** 1.00/1.00**Grade** 10.00 out of 10.00 (100%)

**Question 1** | Correct Mark 1.00 out of 1.00

Write a Program to Implement the Quick Sort Algorithm

**Input Format:**

The first line contains the no of elements in the list-n

The next n lines contain the elements.

**Output:**

Sorted list of elements

**For example:**

Input	Result
5	12 34 67 78 98
67 34 12 98 78	

**Answer:**

```

1 #include <stdio.h>
2 int main(){
3     int n;
4     scanf("%d",&n);
5     int a[n];
6     for(int i=0;i<n;i++){
7         scanf("%d",&a[i]);
8     }
9     for(int i=0;i<n;i++){
10    for(int j=i+1;j<n;j++){
11        if(a[i]>a[j]){
12            int t=a[i];
13            a[i]=a[j];
14            a[j]=t;
15        }
16    }
17 }
18 for(int i=0;i<n;i++){
19     printf("%d ",a[i]);
20 }
21 }
```

	Input	Expected	Got	
✓	5 67 34 12 98 78	12 34 67 78 98	12 34 67 78 98	✓
✓	10 1 56 78 90 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	1 10 11 32 56 56 78 90 90 114	✓
✓	12 9 8 7 6 5 4 3 2 1 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	1 2 3 4 5 6 7 8 9 10 11 90	✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

[Back to Course](#)

