

Full Name:

Remi Chartier

Email:

remipr.chartier@gmail.com

Test Name:

**Mock Test** 

Taken On:

29 Oct 2021 22:34:07 IST

Time Taken:

2 min 17 sec/ 10 min

**Contact Number:** 

+14084751573

Linkedin:

http://www.linkedin.com/in/remichartier

Invited by:

Ankush

Invited on:

29 Oct 2021 22:33:47 IST

Skills Score:

Tags Score:

Algorithms 105/105

Core CS 105/105

Easy 105/105

Problem Solving 105/105

Search 105/105

Sorting 105/105 problem-solving 105/105

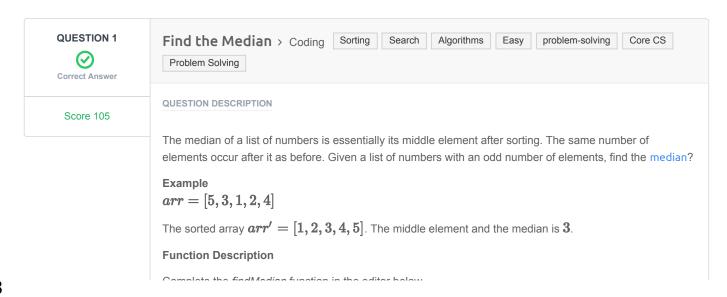
100% 105/105

scored in **Mock Test** in 2 min 17 sec on 29 Oct 2021 22:34:07 IST

#### **Recruiter/Team Comments:**

No Comments.





Complete the *finalitedian* function in the editor below.

findMedian has the following parameter(s):

• int arr[n]: an unsorted array of integers

#### Returns

• int: the median of the array

## **Input Format**

The first line contains the integer n, the size of arr.

The second line contains  $m{n}$  space-separated integers  $m{arr}[i]$ 

## **Constraints**

- $1 \le n \le 1000001$
- n is odd
- $-10000 \le arr[i] \le 10000$

# Sample Input 0

```
7
0 1 2 4 6 5 3
```

## Sample Output 0

3

# **Explanation 0**

The sorted arr = [0, 1, 2, 3, 4, 5, 6]. It's middle element is at arr[3] = 3.

## **CANDIDATE ANSWER**

Language used: C++

```
1  /*
2  * Complete the 'findMedian' function below.
3  *
4  * The function is expected to return an INTEGER.
5  * The function accepts INTEGER_ARRAY arr as parameter.
6  */
7
8 int findMedian(vector<int> arr) {
      sort(arr.begin(),arr.end());
      return arr[int((arr.size() / 2))];
10
11
12
13
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 1	Easy	Sample case	Success	0	0.0247 sec	8.98 KB
Testcase 2	Easy	Hidden case	Success	35	0.0247 sec	9.14 KB
Testcase 3	Easy	Hidden case	Success	35	0.0316 sec	9.18 KB
Testcase 4	Easy	Hidden case	Success	35	0.0568 sec	13.5 KB

No Comments