



Full Name: Shaurya kajiwala

Email: kajiwalashaurya29@gmail.com

Test Name: Mock Test

Taken On: 9 Jun 2023 12:48:34 IST

Time Taken: 8 min 2 sec/ 15 min

Invited by: Ankush

Invited on: 8 Jun 2023 17:13:59 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 8 min 2 sec on 9 Jun 2023 12:48:34 IST

Recruiter/Team Comments:

No Comments.

| | Question Description | Time Taken | Score | Status |
|----|--------------------------|--------------|----------|--------|
| Q1 | Find the Median > Coding | 7 min 37 sec | 105/ 105 | ✓ |

QUESTION 1

✓

Correct Answer

Score 105

Find the Median > Coding

Sorting

Search

Algorithms

Easy

problem-solving

Core CS

Problem Solving

QUESTION DESCRIPTION

The median of a list of numbers is essentially its middle element after sorting. The same number of elements occur after it as before. Given a list of numbers with an odd number of elements, find the **median**?

Example
 $arr = [5, 3, 1, 2, 4]$

The sorted array $arr' = [1, 2, 3, 4, 5]$. The middle element and the median is **3**.

Function Description

Complete the `findMedian` 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 *n* space-separated integers *arr[i]*

Constraints

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

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: Java 8

```
1 class Result {
2
3     /*
4      * Complete the 'findMedian' function below.
5      *
6      * The function is expected to return an INTEGER.
7      * The function accepts INTEGER_ARRAY arr as parameter.
8      */
9
10    public static int findMedian(List<Integer> arr) {
11        Collections.sort(arr);
12        int length=arr.size();
13        int medianElement=Math.floorDiv(length, 2);
14        return arr.get(medianElement);
15    }
16 }
17
18 }
```

| TESTCASE | DIFFICULTY | TYPE | STATUS | SCORE | TIME TAKEN | MEMORY USED |
|------------|------------|-------------|-----------|-------|------------|-------------|
| Testcase 1 | Easy | Sample case | ✔ Success | 0 | 0.2113 sec | 30.1 KB |
| Testcase 2 | Easy | Hidden case | ✔ Success | 35 | 0.1343 sec | 31.3 KB |
| Testcase 3 | Easy | Hidden case | ✔ Success | 35 | 0.1533 sec | 32.3 KB |
| Testcase 4 | Easy | Hidden case | ✔ Success | 35 | 0.3981 sec | 45.2 KB |

