



Full Name: GUILHERME FERNANDES

Email: contato@guifr.com.br

Test Name: Mock Test

Taken On: 17 May 2022 08:14:48 IST

Time Taken: 3 min 17 sec/ 15 min

Invited by: Ankush

Invited on: 17 May 2022 08:14:34 IST

Skills Score:

Tags Score:

- Algorithms105/105
- Core CS105/105
- Easy105/105
- Problem Solving105/105
- Search105/105
- Sorting105/105
- problem-solving105/105

100%

105/105

scored in Mock Test in 3 min 17 sec on 17 May 2022 08:14:48 IST

Recruiter/Team Comments:

No Comments.

	Question Description	Time Taken	Score	Status
Q1	Find the Median > Coding	2 min 59 sec	105/ 105	

QUESTION 1

Correct Answer

Score 105

Find the Median > Coding

SortingSearchAlgorithmsEasyproblem-solvingCore 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        int median = arr.size() >> 1;
12        Collections.sort(arr);
13        return arr.get(median);
14    }
15
16 }
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 1	Easy	Sample case	✔ Success	0	0.1649 sec	30 KB
Testcase 2	Easy	Hidden case	✔ Success	35	0.1851 sec	31.7 KB
Testcase 3	Easy	Hidden case	✔ Success	35	0.2544 sec	32.3 KB
Testcase 4	Easy	Hidden case	✔ Success	35	0.4853 sec	44.8 KB

No Comments

