

#### Mock Test > herrera.fernando1209@gmail.com

Full Name:

Fernando Herrera

Email:

herrera.fernando1209@gmail.com

Test Name:

**Mock Test** 

Taken On:

23 Jan 2024 04:47:23 IST

10 min 37 sec/ 15 min

Time Taken: Linkedin:

https://www.linkedin.com/in/fernandoherrera2190/

Invited by:

Ankush

Invited on:

23 Jan 2024 04:37:58 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

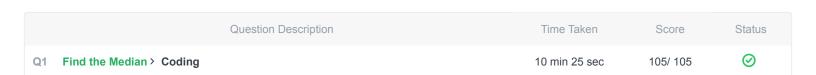
problem-solving 105/105

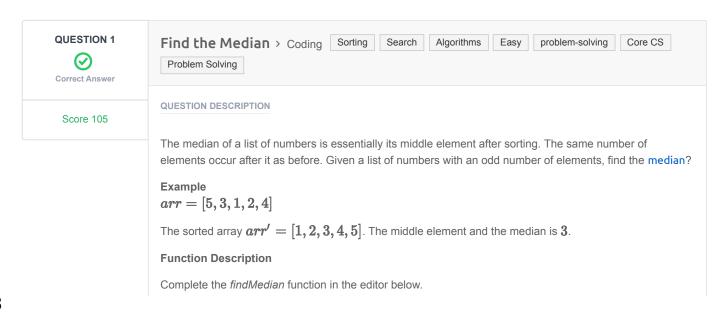
100% 105/105

scored in **Mock Test** in 10 min 37 sec on 23 Jan 2024 04:47:23 IST

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

No Comments.





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  $m{arr}[m{i}]$ 

#### **Constraints**

- $1 \le n \le 1000001$
- $oldsymbol{\cdot}$  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: JavaScript (Node.js)

```
1 'use strict';
 3 const fs = require('fs');
 5 process.stdin.resume();
 6 process.stdin.setEncoding('utf-8');
8 let inputString = '';
9 let currentLine = 0;
11 process.stdin.on('data', function (inputStdin) {
       inputString += inputStdin;
13 });
14
15 process.stdin.on('end', function () {
     inputString = inputString.split('\n');
      main();
19 });
21 function readLine() {
     return inputString[currentLine++];
23 }
24
* Complete the 'findMedian' function below.
```

```
* The function is expected to return an INTEGER.
31 * The function accepts INTEGER ARRAY arr as parameter.
32 */
34 function findMedian(arr) {
35 // Write your code here
      arr.sort((a, b) => (a-b))
       console.log(arr)
       return arr[(arr.length - 1 )/ 2]
41 }
43 function main() {
       const ws = fs.createWriteStream(process.env.OUTPUT PATH);
46
      const n = parseInt(readLine().trim(), 10);
47
48
       const arr = readLine().replace(/\s+$/g, '').split(' ').map(arrTemp =>
49 parseInt(arrTemp, 10));
       const result = findMedian(arr);
       ws.write(result + '\n');
       ws.end();
56 }
  TESTCASE
            DIFFICULTY
                            TYPE
                                       STATUS
                                                 SCORE
                                                         TIME TAKEN
                                                                     MEMORY USED
  Testcase 1
                Easy
                         Sample case
                                     Success
                                                          0.0378 sec
                                                                         42 KB
  Testcase 2
                Easy
                         Hidden case
                                     Success
                                                          0.0411 sec
                                                                        43.7 KB
  Testcase 3
                Easy
                         Hidden case
                                     Success
                                                   35
                                                          0.048 sec
                                                                        44.9 KB
                                                          0.0853 sec
                                                                        55.5 KB
  Testcase 4
                Easy
                         Hidden case
                                     Success
                                                   35
No Comments
```

PDF generated at: 22 Jan 2024 23:29:32 UTC