Full Name:	Md Shafiul Islam
Email:	mdshafiulmondol02@gmail.com
Test Name:	**Mock Test**
Taken On:	27 Dec 2024 18:59:30 IST
Time Taken:	12 min 2 sec/ 15 min
Invited by:	Ankush
Invited on:	27 Dec 2024 18:58:42 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 12 min 2 sec on 27 Dec 2024 18:59:30 IST

Recruiter/Team Comments:

No Comments.

	Question Description	Time Taken	Score	Status
Q1	Find the Median > **Coding**	11 min 39 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: **C++14**

```
#include <bits/stdc++.h>
using namespace std;
int main(){
    int n,a;
    cin>>n;
     priority_queue<int, vector<int>, greater<int>> p;
    vector <int >v;
    for (int i=0;i<n;i++){
        cin>>a;
        p.push(a);
    }
    while(!p.empty()){
        v.push_back(p.top());
        p.pop();
    }
    cout<<v[n/2];
}
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 1	Easy	Sample case	Success	0	0.0087 sec	8.83 KB
Testcase 2	Easy	Hidden case	Success	35	0.0113 sec	9.01 KB
Testcase 3	Easy	Hidden case	Success	35	0.0113 sec	9.05 KB
Testcase 4	Easy	Hidden case	Success	35	0.0396 sec	9.1 KB

No Comments