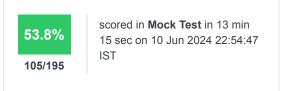


Mock Test > juanserrato100@gmail.com

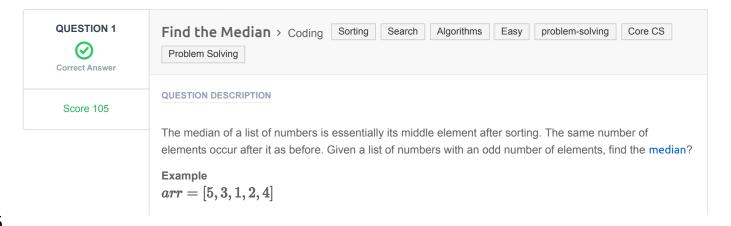
Full Name: Juan Serrato Email: juanserrato100@gmail.com Test Name: **Mock Test** Taken On: 10 Jun 2024 22:54:47 IST Time Taken: 13 min 15 sec/ 40 min Invited by: Ankush Invited on: 10 Jun 2024 22:54:09 IST Skills Score: Tags Score: Algorithms 105/195 Constructive Algorithms 0/90 Core CS 105/195 Easy 105/105 Greedy Algorithms 0/90 Medium 0/90 Problem Solving 105/195 105/105 Search Sorting 105/105 problem-solving 105/195



Recruiter/Team Comments:

No Comments.





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

- $1 \le n \le 1000001$
- 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: JavaScript (Node.js)

```
1
2  /*
3  * Complete the 'findMedian' function below.
4  *
5  * The function is expected to return an INTEGER.
6  * The function accepts INTEGER_ARRAY arr as parameter.
7  */
8
9 function findMedian(arr) {
    // Write your code here
    const sortedArray = arr.sort((a,b) => a - b);
    return sortedArray[Math.trunc(sortedArray.length / 2)];
13
14 }
15
```

TESTCASE	DIFFICULTY	TYPE	STATUS	SCORE	TIME TAKEN	MEMORY USED
Testcase 1	Easy	Sample case	Success	0	0.0707 sec	41.8 KB
Testcase 2	Easy	Hidden case	Success	35	0.0448 sec	43.1 KB
Testcase 3	Easy	Hidden case	Success	35	0.0467 sec	44.6 KB
Testcase 4	Easv	Hidden case	Success	35	0.099 sec	55.3 KB



QUESTION 2



Score 0



QUESTION DESCRIPTION

Sean invented a game involving a $2n \times 2n$ matrix where each cell of the matrix contains an integer. He can reverse any of its rows or columns any number of times. The goal of the game is to maximize the sum of the elements in the $n \times n$ submatrix located in the upper-left quadrant of the matrix.

Given the initial configurations for q matrices, help Sean reverse the rows and columns of each matrix in the best possible way so that the sum of the elements in the matrix's upper-left quadrant is maximal.

Example

$$matrix = [[1, 2], [3, 4]]$$

1 2

3 4

It is 2×2 and we want to maximize the top left quadrant, a 1×1 matrix. Reverse row 1:

1 2

4 3

And now reverse column 0:

4 2

1 3

The maximal sum is 4.

Function Description

Complete the flippingMatrix function in the editor below.

flippingMatrix has the following parameters:

- int matrix[2n][2n]: a 2-dimensional array of integers

Returns

- int: the maximum sum possible.

Input Format

The first line contains an integer \boldsymbol{q} , the number of queries.

The next q sets of lines are in the following format:

- The first line of each query contains an integer, n.
- Each of the next 2n lines contains 2n space-separated integers matrix[i][j] in row i of the matrix.

Constraints

- $1 \le q \le 16$
- $1 \le n \le 128$
- $0 \leq matrix[i][j] \leq 4096$, where $0 \leq i, j < 2n$.

Sample Input

```
STDIN
            Function
            q = 1
   n = 2
112 42 83 119 matrix = [[112, 42, 83, 119], [56, 125, 56, 49], \
56 125 56 49 [15, 78, 101, 43], [62, 98, 114, 108]]
15 78 101 43
62 98 114 108
```

Sample Output

414

Explanation

Start out with the following $2n \times 2n$ matrix:

$$matrix = egin{bmatrix} 112 & 42 & 83 & 119 \ 56 & 125 & 56 & 49 \ 15 & 78 & 101 & 43 \ 62 & 98 & 114 & 108 \ \end{bmatrix}$$

Perform the following operations to maximize the sum of the $n \times n$ submatrix in the upper-left quadrant:

2. Reverse column 2 ([83, 56, 101, 114] ightarrow [114, 101, 56, 83]), resulting in the matrix:

$$matrix = egin{bmatrix} 112 & 42 & 114 & 119 \ 56 & 125 & 101 & 49 \ 15 & 78 & 56 & 43 \ 62 & 98 & 83 & 108 \ \end{bmatrix}$$

3. Reverse row 0 ([112, 42, 114, 119] \rightarrow [119, 114, 42, 112]), resulting in the matrix:

$$matrix = \begin{bmatrix} 119 & 114 & 42 & 112 \\ 56 & 125 & 101 & 49 \\ 15 & 78 & 56 & 43 \\ 62 & 98 & 83 & 108 \end{bmatrix}$$

The sum of values in the n imes n submatrix in the upper-left quadrant is 119+114+56+125=414

CANDIDATE ANSWER

No answer was submitted for this question. Showing compiled/saved versions.

Language used: JavaScript (Node.js)

```
2 /*
  * Complete the 'flippingMatrix' function below.
4 *
5 * The function is expected to return an INTEGER.
  * The function accepts 2D INTEGER ARRAY matrix as parameter.
7 */
9 function flippingMatrix(matrix) {
   // Write your code here
    let result = 0;
     let halfMatrixSize = matrix.size() / 2;
        let maxMatrixIndex = matrix.size() - 1;
```

```
14
           for (let row = 0; row < halfMatrixSize; row ++) {</pre>
               for (let col = 0; col < halfMatrixSize; col ++) {</pre>
                   let element = matrix.get(row).get(col);
                   let rightMirror = matrix.get(row).get(maxMatrixIndex - col);
                   let downMirror = matrix.get(maxMatrixIndex - row).get(col);
                   let rightDownMirror = matrix.get(maxMatrixIndex - row)
                   .get(maxMatrixIndex - col);
                   result += Arrays.asList(element, rightMirror, downMirror,
24 rightDownMirror)
                   .stream().max(Integer::compare).get();
              }
          }
          return result;
30 }
No Comments
```

PDF generated at: 10 Jun 2024 18:06:58 UTC