Search

Ander Cabrera
@andercabrerac

My Badges

Problem Solving
★★★

Personal Information

✉ andercabrerac@gmail.com

☎ Add your mobile number

◎ Guatemala

My Certifications

You have not earned any certificates yet. Get Certified

Work Experience   + Add Work Experience

Add your work experience. Don't forget to add those internships as well.

My Resume   + Add Resume

Submissions

Problem	Language	Time	Result	Score	Action
Matrix Layer Rotation	javascript	5 minutes	Accepted	80.0	View results
Two Two	javascript	about 1 hour	Accepted	150.0	View results
Two Two	javascript	about 1 hour	Wrong Answer	5.56	View results
Matrix Layer Rotation	javascript	about 1 hour	Wrong Answer	53.33	View results
Matrix Layer Rotation	javascript	about 1 hour	Wrong Answer	53.33	View results
Organizing Containers of Balls	javascript	about 2 hours	Accepted	30.0	View results
Organizing Containers of Balls	javascript	about 2 hours	Accepted	30.0	View results
Forming a Magic Square	javascript	about 3 hours	Accepted	20.0	View results

Forming a Magic Square	javascript	about 3 hours	Accepted	20.0	View results
Number Line Jumps	javascript	7 days	Accepted	10.0	View results

Problem	Language	Time	Result	Score	Action
Number Line Jumps	javascript	7 days	Terminated due to timeout	4.05	View results
Number Line Jumps	javascript	7 days	Terminated due to timeout	4.05	View results
Mini-Max Sum	typescript	7 days	Accepted	10.0	View results
Staircase	typescript	7 days	Accepted	10.0	View results
Plus Minus	typescript	7 days	Accepted	10.0	View results
Simple Array Sum	typescript	7 days	Accepted	10.0	View results

EASY

Plus Minus

Staircase

Mini-max-sum

Kangaroo


```
26   * Complete the 'kangaroo' function below.
27   *
28   * The function is expected to return a STRING.
29   * The function accepts following parameters:
30   *  1. INTEGER x1
31   *  2. INTEGER v1
32   *  3. INTEGER x2
33   *  4. INTEGER v2
34   */
35
36 v function kangaroo(x1, v1, x2, v2) {
37 v   if (x2 > x1 && v2 >= v1) {
38       return "NO";
39     }
40 v   else if ((x2 - x1) % (v1 - v2) === 0) {
41       return "YES";
42     }
43 v   else {
44       return "NO";
45     }
46   }
47
48 v function main() {
49     const ws = fs.createWriteStream(process.env.OUTPUT_PATH);
50
51     const firstMultipleInput = readLine().replace(/\s+$/g, '').split(' ');
52
                                                       Line: 67 Col: 1
```

MEDIUM

Organizing Containers of Balls


```javascript
26    * Complete the 'organizingContainers' function below.
27    *
28    * The function is expected to return a STRING.
29    * The function accepts 2D_INTEGER_ARRAY container as parameter.
30    */
31
32  function organizingContainers(container) {
33      // Write your code here
34      let a = new Array(container.length).fill(0);
35      let b = new Array(container.length).fill(0);
36
37      for (let i = 0; i < container.length; i++) {
38          for (let j = 0; j < container.length; j++) {
39              a[i] += container[i][j];
40              b[j] += container[i][j];
41          }
42      }
43      a.sort((a, b) => a - b);
44      b.sort((a, b) => a - b);
45
46      for (let i = 0; i < container.length; i++) {
47          if (a[i] != b[i]) {
48              return "Impossible";
49          }
50      }
51      return "Possible";
52  }
```

Magic Square

Prepare > Algorithms > Implementation > Forming a Magic Square > Submissions

Forming a Magic Square ★

171.67 more points to get your next star!

Rank: 685951 | Points: 303.33/475

Problem	Submissions	Leaderboard	Discussions	Editorial

RESULT	SCORE	LANGUAGE	TIME	
✓ Accepted	20.0	Javascript	2 hours ago	View Results
✓ Accepted	20.0	Javascript	3 hours ago	View Results

‹ 1 ›

NEED HELP?

View discussions

View editorial

View top submissions

Change Theme Language JavaScript (Node.js)

```javascript
32  function formingMagicSquare(s) {
33    const magicSquares = [
34      [[8, 1, 6], [3, 5, 7], [4, 9, 2]],
35      [[6, 1, 8], [7, 5, 3], [2, 9, 4]],
36      [[4, 9, 2], [3, 5, 7], [8, 1, 6]],
37      [[2, 9, 4], [7, 5, 3], [6, 1, 8]],
38      [[8, 3, 4], [1, 5, 9], [6, 7, 2]],
39      [[4, 3, 8], [9, 5, 1], [2, 7, 6]],
40      [[6, 7, 2], [1, 5, 9], [8, 3, 4]],
41      [[2, 7, 6], [9, 5, 1], [4, 3, 8]]
42    ];
43
44    let minCost = Infinity;
45
46    for (const magicSquare of magicSquares) {
47      let cost = 0;
48
49      for (let i = 0; i < 3; i++) {
50        for (let j = 0; j < 3; j++) {
51          cost += Math.abs(s[i][j] - magicSquare[i][j]);
52        }
53      }
54
55      minCost = Math.min(minCost, cost);
56    }
57
58    return minCost;
```

Line: 76 Col: 1

Upload Code as File   Test against custom input   Run Code   Submit Code

HARD

Matrix Layer Rotation

Two-Two