

LeetCode

Problem List

Premium

🕒

🔒

👤

Description

Discussion (29)

Solutions (2.9K)

Submissions

931. Minimum Falling Path Sum

Medium

👍 4.3K

👎 117

☆

🔗

Companies

Given an $n \times n$ array of integers *matrix*, return the *minimum sum of any falling path through matrix*.

A *falling path* starts at any element in the first row and chooses the element in the next row that is either directly below or diagonally left/right. Specifically, the next element from position (row, col) will be $(row + 1, col - 1)$, $(row + 1, col)$, or $(row + 1, col + 1)$.

Example 1:

2	1	3
6	5	4
7	8	9

2	1	3
6	5	4
7	8	9

Input: matrix = [[2,1,3],[6,5,4],[7,8,9]]

Output: 13

Explanation: There are two falling paths with a minimum sum as shown.

Example 2:

-19	57
-40	-5

-19	57
-40	-5

Input: matrix = [[-19,57],[-40,-5]]

Output: -59

Explanation: The falling path with a minimum sum is shown.

Constraints:

- $n == matrix.length == matrix[i].length$
- $1 \leq n \leq 100$
- $-100 \leq matrix[i][j] \leq 100$

Accepted 222.3K

Submissions 321.6K

Acceptance Rate 69.1%

Seen this question in a real interview before?

1/4

Yes

No

Similar Questions

Related Topics

Java

Auto

```
1 class Solution {
2     public int minFallingPathSum(int[][] matrix) {
3
4     }
5 }
```

Console

Run

Submit

https://leetcode.com/problems/minimum-falling-path-sum/description/

1/341