

Problem 3341: Find Minimum Time to Reach Last Room I

Problem Information

Difficulty: Medium

Acceptance Rate: 55.53%

Paid Only: No

Tags: Array, Graph, Heap (Priority Queue), Matrix, Shortest Path

Problem Description

There is a dungeon with $n \times m$ rooms arranged as a grid.

You are given a 2D array `moveTime` of size $n \times m$, where `moveTime[i][j]` represents the **minimum** time in seconds **after** which the room opens and can be moved to. You start from the room $(0, 0)$ at time $t = 0$ and can move to an **adjacent** room. Moving between adjacent rooms takes exactly one second.

Return the **minimum** time to reach the room $(n - 1, m - 1)$.

Two rooms are **adjacent** if they share a common wall, either horizontally or vertically.

Example 1:

Input: `moveTime = [[0,4],[4,4]]`

Output: 6

Explanation:

The minimum time required is 6 seconds.

* At time $t == 4$, move from room $(0, 0)$ to room $(1, 0)$ in one second. * At time $t == 5$, move from room $(1, 0)$ to room $(1, 1)$ in one second.

****Example 2:****

****Input:**** moveTime = [[0,0,0],[0,0,0]]

****Output:**** 3

****Explanation:****

The minimum time required is 3 seconds.

* At time $t == 0$, move from room $(0, 0)$ to room $(1, 0)$ in one second. * At time $t == 1$, move from room $(1, 0)$ to room $(1, 1)$ in one second. * At time $t == 2$, move from room $(1, 1)$ to room $(1, 2)$ in one second.

****Example 3:****

****Input:**** moveTime = [[0,1],[1,2]]

****Output:**** 3

****Constraints:****

* $2 \leq n \leq \text{moveTime.length} \leq 50$ * $2 \leq m \leq \text{moveTime}[i].\text{length} \leq 50$ * $0 \leq \text{moveTime}[i][j] \leq 109$

Code Snippets

C++:

```
class Solution {
public:
    int minTimeToReach(vector<vector<int>>& moveTime) {

    }
};
```

Java:

```
class Solution {  
    public int minTimeToReach(int[][] moveTime) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def minTimeToReach(self, moveTime: List[List[int]]) -> int:
```