Problem 407: Trapping Rain Water II

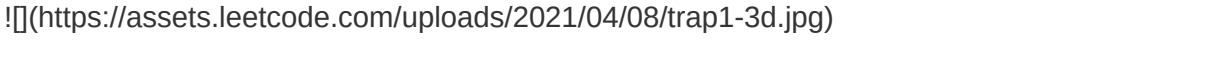
Problem Information

Difficulty: Hard
Acceptance Rate: 63.91%
Paid Only: No
Tags: Array, Breadth-First Search, Heap (Priority Queue), Matrix

Problem Description

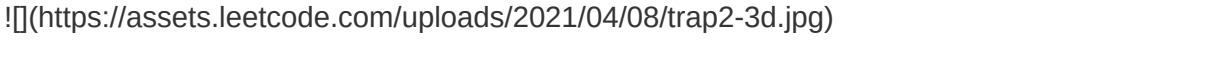
Given an `m x n` integer matrix `heightMap` representing the height of each unit cell in a 2D elevation map, return _the volume of water it can trap after raining_.

Example 1:



Input: heightMap = [[1,4,3,1,3,2],[3,2,1,3,2,4],[2,3,3,2,3,1]] **Output:** 4 **Explanation:** After the rain, water is trapped between the blocks. We have two small ponds 1 and 3 units trapped. The total volume of water trapped is 4.

Example 2:



Input: heightMap = [[3,3,3,3,3],[3,2,2,2,3],[3,2,1,2,3],[3,2,2,2,3],[3,3,3,3,3]] **Output:** 10

Constraints:

* `m == heightMap.length` * `n == heightMap[i].length` * `1 <= m, n <= 200` * `0 <= heightMap[i][j] <= 2 * 104`

Code Snippets

C++:

```
class Solution {
public:
int trapRainWater(vector<vector<int>>& heightMap) {

}
};
```

Java:

```
class Solution {
public int trapRainWater(int[][] heightMap) {

}
}
```

Python3:

```
class Solution:
def trapRainWater(self, heightMap: List[List[int]]) -> int:
```