417. Pacific Atlantic Water Flow

Medium

There is an m x n rectangular island that borders both the Pacific Ocean and Atlantic Ocean. The Pacific Ocean touches the island's left and top edges, and the Atlantic Ocean touches the island's right and bottom edges.

The island is partitioned into a grid of square cells. You are given an m x n integer matrix heights where heights[r][c] represents the height above sea level of the cell at coordinate (r, c).

The island receives a lot of rain, and the rain water can flow to neighboring cells directly north, south, east, and west if the neighboring cell's height is less than or equal to the current cell's height. Water can flow from any cell adjacent to an ocean into the ocean.
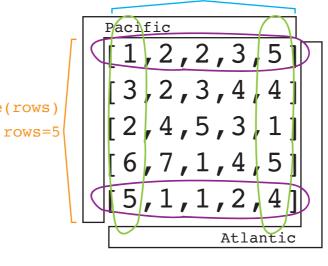
Return a 2D list of grid coordinates result where result[i] = [ri, ci] denotes that rain water can flow from cell (ri, ci) to both the Pacific and Atlantic oceans.

```
class Solution:
    def pacificAtlantic(self, heights: List[List[int]]) -> List[List[int]]:

        ROWS, COLS = len(heights), len(heights[0])
        pacific, atlantic = set(), set()

        def dfs(r, c, visit, prevHeight):
            if (
                (r, c) in visit
                or r < 0
                or c < 0
                or r == ROWS
                or c == COLS
                or heights[r][c] < prevHeight
            ):
                return
            visit.add((r, c))
            dfs(r + 1, c, visit, heights[r][c])
            dfs(r - 1, c, visit, heights[r][c])
            dfs(r, c + 1, visit, heights[r][c])
            dfs(r, c - 1, visit, heights[r][c])

        for c in range(COLS):
            dfs(0, c, pacific, heights[0][c])
            dfs(ROWS - 1, c, atlantic, heights[ROWS - 1][c])

        for r in range(ROWS):
            dfs(r, 0, pacific, heights[r][0])
            dfs(r, COLS - 1, atlantic, heights[r][COLS - 1])

        result = []
        for r in range(ROWS):
            for c in range(COLS):
                if (r, c) in pacific and (r, c) in atlantic:
                    result.append([r, c])
        return result
```

Input: heights =
[[1,2,2,3,5],[3,2,3,4,4],[2,4,5,3,1],[6,7,1,4,5],[5,1,1,2,4]]
Output: [[0,4],[1,3],[1,4],[2,2],[3,0],[3,1],[4,0]]
Explanation: The following cells can flow to the Pacific and Atlantic oceans, as shown below:
[0,4]: [0,4] -> Pacific Ocean
       [0,4] -> Atlantic Ocean
[1,3]: [1,3] -> [0,3] -> Pacific Ocean
       [1,3] -> [1,4] -> Atlantic Ocean
[1,4]: [1,4] -> [1,3] -> [0,3] -> Pacific Ocean
       [1,4] -> Atlantic Ocean
[2,2]: [2,2] -> [1,2] -> [0,2] -> Pacific Ocean
       [2,2] -> [2,3] -> [2,4] -> Atlantic Ocean
[3,0]: [3,0] -> Pacific Ocean
       [3,0] -> [4,0] -> Atlantic Ocean
[3,1]: [3,1] -> [3,0] -> Pacific Ocean
       [3,1] -> [4,1] -> Atlantic Ocean
[4,0]: [4,0] -> Pacific Ocean
       [4,0] -> Atlantic Ocean
Note that there are other possible paths for these cells to flow to the Pacific and Atlantic oceans.

c in range(cols)
cols=5

Pacific

```
[1,2,2,3,5]
[3,2,3,4,4]
[2,4,5,3,1]
[6,7,1,4,5]
[5,1,1,2,4]
```

r in range(rows)
rows=5

Atlantic

dfs(0, 0, pacific, heights[0][0])
dfs(0, ~~COLS - 1~~ 4, atlantic, heights[0][~~COLS - 1~~ 4])

dfs(1, 0, pacific, heights[1][0])
dfs(1, ~~COLS - 1~~ 4, atlantic, heights[1][~~COLS - 1~~ 4])

dfs(2, 0, pacific, heights[2][0])
dfs(2, ~~COLS - 1~~ 4, atlantic, heights[2][~~COLS - 1~~ 4])

dfs(3, 0, pacific, heights[3][0])
dfs(3, ~~COLS - 1~~ 4, atlantic, heights[3][~~COLS - 1~~ 4])

dfs(4, 0, pacific, heights[4][0])
dfs(4, ~~COLS - 1~~ 4, atlantic, heights[4][~~COLS - 1~~ 4])

dfs(0, 0, pacific, heights[0][0])
dfs(~~ROWS - 1~~ 4, 0, atlantic, heights[~~ROWS - 1~~ 4][0])

dfs(0, 1, pacific, heights[0][1])
dfs(~~ROWS - 1~~ 4, 1, atlantic, heights[~~ROWS - 1~~ 4][1])

dfs(0, 2, pacific, heights[0][2])
dfs(~~ROWS - 1~~ 4, 2, atlantic, heights[~~ROWS - 1~~ 4][2])

dfs(0, 3, pacific, heights[0][3])
dfs(~~ROWS - 1~~ 4, 3, atlantic, heights[~~ROWS - 1~~ 4][3])

dfs(0, 4, pacific, heights[0][4])
dfs(~~ROWS - 1~~ 4, 4, atlantic, heights[~~ROWS - 1~~ 4][4])