417. Pacific Atlantic Water Flow

QuestionEditorial Solution

Given an m x n matrix of non-negative integers representing the height of each unit cell in a continent, the "Pacific ocean" touches the left and top edges of the matrix and the "Atlantic ocean" touches the right and bottom edges.

Water can only flow in four directions (up, down, left, or right) from a cell to another one with height equal or lower.

Find the list of grid coordinates where water can flow to both the Pacific and Atlantic ocean.

Note:

1. The order of returned grid coordinates does not matter.

2. Both *m* and *n* are less than 150.

```
Given the following 5x5 matrix:

  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   *
          *   *    *    *    *  Atlantic

Return:

[[0, 4], [1, 3], [1, 4], [2, 2], [3, 0], [3, 1], [4, 0]] (positions with parentheses in above matrix).
```

```cpp
class Solution {
public:
    vector<pair<int, int>> pacificAtlantic(vector<vector<int>>& matrix) {
        int h = matrix.size();
        if(!h)return {};
        int w = matrix[0].size();

        vector<vector<int>> vv(h,vector<int>(w,0));
        for(int i=0;i<h;++i) {
            explore<1>(matrix,vv,i,0,INT_MIN);// left
            explore<2>(matrix,vv,i,w-1,INT_MIN);// right
        }
```

```cpp
        for(int i=0;i<w;++i) {
            explore<1>(matrix,vv,0,i,INT_MIN);//top
            explore<2>(matrix,vv,h-1,i,INT_MIN);//buttom
        }
        vector<pair<int,int>> ret;
        for(int i=0;i<h;++i) for(int j=0;j<w;++j) if(vv[i][j] ==
3)ret.push_back({i,j});
        return ret;
    }

    //(zhewei): check all the coordinate which satisfies both mask1 and mask2
    template<int MASK>
    void explore(const vector<vector<int>>& matrix, vector<vector<int>>& vv, int x, int
y, int h) {
        if(x<0 || y<0 || x>=matrix.size() || y>=matrix[x].size())return;
        if(vv[x][y]&MASK) return;
        if(h > matrix[x][y]) return;
        vv[x][y] |= MASK;
        h = matrix[x][y];
        explore<MASK>(matrix,vv,x-1,y,h);
        explore<MASK>(matrix,vv,x+1,y,h);
        explore<MASK>(matrix,vv,x,y-1,h);
        explore<MASK>(matrix,vv,x,y+1,h);
    }
};
```