

Description

Solution

Discuss (645)

Submissions

C++

Auto

694. Number of Distinct Islands

Medium

1988

124

Add to List

Share

You are given an $m \times n$ binary matrix `grid`. An island is a group of 1's (representing land) connected **4-directionally** (horizontal or vertical.) You may assume all four edges of the grid are surrounded by water.

An island is considered to be the same as another if and only if one island can be translated (and not rotated or reflected) to equal the other.

Return the number of **distinct** islands.

Example 1:

1	1	0	0	0
1	1	0	0	0
0	0	0	1	1
0	0	0	1	1

Input: `grid = [[1,1,0,0,0],[1,1,0,0,0],[0,0,0,1,1],[0,0,0,1,1]]`

Output: 1

Example 2:

```

1  class Solution
2  public:
3      bool valid(int x, int y, int m, int n, vector<vector<int>>& grid) {
4          if(x < 0 || x >= m || y < 0 || y >= n || grid[x][y] == 0) return false;
5          return true;
6      }
7
8      void dfs(int x, int y, vector<vector<int>>& grid, string &str) {
9          grid[x][y] = 2;
10         str += "1";
11         int dx[] = {0, 1, 0, -1, 0};
12         int dy[] = {1, 0, 1, 0, -1};
13         for(int i = 0; i < 4; i++) {
14             int nx = x + dx[i];
15             int ny = y + dy[i];
16             if(valid(nx, ny, grid.size(), grid[0].size(), grid)) {
17                 dfs(nx, ny, grid, str);
18             }
19         }
20         str += "0";
21     }
22
23     int numDistinctIslands(vector<vector<int>>& grid) {
24         int m = grid.size();
25         int n = grid[0].size();
26         unordered_set<string> islands;
27         for(int i = 0; i < m; i++) {
28             for(int j = 0; j < n; j++) {
29                 if(grid[i][j] == 1) {
30                     dfs(i, j, grid, "");
31                     islands.insert(str);
32                 }
33             }
34         }
35         return islands.size();
36     }
37 }
```

Problems

Pick One

< Prev

1/19

Next >

Contribute

Run Code