

Problem 200: Number of Islands

Problem Information

Difficulty: Medium

Acceptance Rate: 63.30%

Paid Only: No

Tags: Array, Depth-First Search, Breadth-First Search, Union Find, Matrix

Problem Description

Given an $m \times n$ 2D binary grid `grid` which represents a map of `'1'`'s (land) and `'0'`'s (water), return `_the number of islands_`.

An **island** is surrounded by water and is formed by connecting adjacent lands horizontally or vertically. You may assume all four edges of the grid are all surrounded by water.

Example 1:

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

Example 2:

Input: `grid = [["1","1","0","0","0"], ["1","1","0","0","0"], ["0","0","1","0","0"], ["0","0","0","1","1"]]`
Output: 3

Constraints:

`m == grid.length` `n == grid[i].length` `1 <= m, n <= 300` `grid[i][j]` is `'0'` or `'1'`.

Code Snippets

C++:

```
class Solution {  
public:  
    int numIslands(vector<vector<char>>& grid) {  
  
    }  
};
```

Java:

```
class Solution {  
    public int numIslands(char[][] grid) {  
  
    }  
}
```

Python3:

```
class Solution:  
    def numIslands(self, grid: List[List[str]]) -> int:
```