Problem 695: Max Area of Island

Problem Information

Difficulty: Medium
Acceptance Rate: 73.57%
Paid Only: No
Tags: Array, Depth-First Search, Breadth-First Search, Union Find, Matrix

Problem Description

You are given an `m x 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.

The **area** of an island is the number of cells with a value `1` in the island.

Return _the maximum**area** of an island in _`grid`. If there is no island, return `0`.

Example 1:

Input: grid = [[0,0,1,0,0,0,0,1,0,0,0,0,0],[0,0,0,0,0,0,0,1,1,1,0,0,0],[0,1,1,0,1,0,0,0,0,0,0,0,0],[0,1,0,0,1,1,0,0,1,0,1,0,0],[0,1,0,0,1,1,0,0,1,1,1,0,0],[0,0,0,0,0,0,0,0,0,0,1,0,0],[0,0,0,0,0,0,0,1,1,1,0,0,0],[0,0,0,0,0,0,0,1,1,0,0,0,0]] **Output:** 6 **Explanation:** The answer is not 11, because the island must be connected 4-directionally.

Example 2:

Input: grid = [[0,0,0,0,0,0,0,0]] **Output:** 0

Constraints:

* `m == grid.length` * `n == grid[i].length` * `1 <= m, n <= 50` * `grid[i][j]` is either `0` or `1`.

Code Snippets

C++:

```
class Solution {
public:
int maxAreaOfIsland(vector<vector<int>>& grid) {

}
};
```

Java:

```
class Solution {
public int maxAreaOfIsland(int[][] grid) {

}
}
```

Python3:

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