Problem 827: Making A Large Island

Problem Information

Difficulty: Hard
Acceptance Rate: 55.91%
Paid Only: No
Tags: Array, Depth-First Search, Breadth-First Search, Union Find, Matrix

Problem Description

You are given an `n x n` binary matrix `grid`. You are allowed to change **at most one** `0` to be `1`.

Return _the size of the largest**island** in_ `grid` _after applying this operation_.

An **island** is a 4-directionally connected group of `1`s.

Example 1:

Input: grid = [[1,0],[0,1]] **Output:** 3 **Explanation:** Change one 0 to 1 and connect two 1s, then we get an island with area = 3.

Example 2:

Input: grid = [[1,1],[1,0]] **Output:** 4 **Explanation:** Change the 0 to 1 and make the island bigger, only one island with area = 4.

Example 3:

Input: grid = [[1,1],[1,1]] **Output:** 4 **Explanation:** Can't change any 0 to 1, only one island with area = 4.

Constraints:

* `n == grid.length` * `n == grid[i].length` * `1 <= n <= 500` * `grid[i][j]` is either `0` or `1`.

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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