Problem 934: Shortest Bridge

Problem Information

Difficulty: Medium
Acceptance Rate: 59.06%
Paid Only: No
Tags: Array, Depth-First Search, Breadth-First Search, Matrix

Problem Description

You are given an `n x n` binary matrix `grid` where `1` represents land and `0` represents water.

An **island** is a 4-directionally connected group of `1`'s not connected to any other `1`'s. There are **exactly two islands** in `grid`.

You may change `0`'s to `1`'s to connect the two islands to form **one island**.

Return _the smallest number of_`0` _' s you must flip to connect the two islands_.

Example 1:

Input: grid = [[0,1],[1,0]] **Output:** 1

Example 2:

Input: grid = [[0,1,0],[0,0,0],[0,0,1]] **Output:** 2

Example 3:

Input: grid = [[1,1,1,1,1],[1,0,0,0,1],[1,0,1,0,1],[1,0,0,0,1],[1,1,1,1,1]] **Output:** 1

Constraints:

* `n == grid.length == grid[i].length` * `2 <= n <= 100` * `grid[i][j]` is either `0` or `1`. * There are exactly two islands in `grid`.

Code Snippets

C++:

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

}
};
```

Java:

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

}
}
```

Python3:

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