

130. Surrounded Regions

Solved ✓

Medium

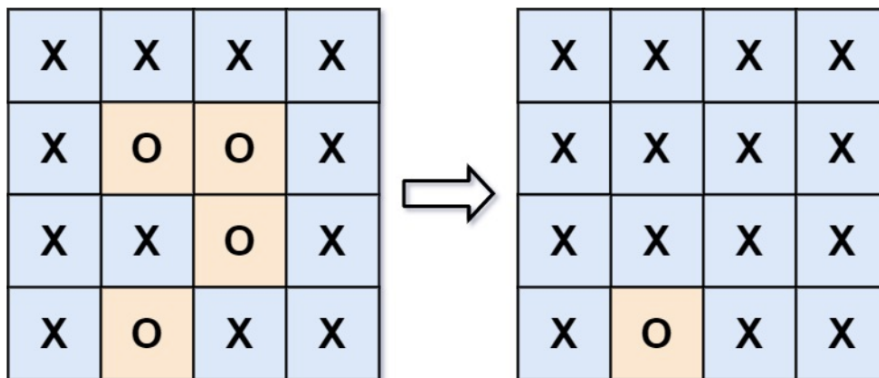
Topics

Companies

Given an $m \times n$ matrix `board` containing `'X'` and `'O'`, capture all regions that are 4-directionally surrounded by `'X'`.

A region is **captured** by flipping all `'O'` s into `'X'` s in that surrounded region.

Example 1:



Code

```
class Solution {
    func solve(_ board: inout [[Character]]) {
        let totalRows = board.count - 1
        let totalColumns = board[0].count - 1

        func dfs(_ r: Int, _ c: Int) {
            if ((r < 0 || r > totalRows || c < 0 || c > totalColumns)) {
                return
            }

            if (board[r][c] == "X" || board[r][c] == "P") { return }

            board[r][c] = "P"
            dfs(r+1,c)
            dfs(r-1,c)
            dfs(r,c+1)
            dfs(r,c-1)
        }
    }
}
```

```

        // first convert outer 0 to T
        for i in (0...totalRows) {
            for j in (0...totalColumns) {
                if(i == 0 || i == totalRows || j == 0 || j == totalColumns) &&
board[i][j] == "O" {
                    dfs(i,j)
                }
            }
        }
        // convert O to X
        for i in (0...totalRows) {
            for j in (0...totalColumns) {
                if(board[i][j] == "O") {
                    board[i][j] = "X"
                }
            }
        }
        // convert P to O
        for i in (0...totalRows) {
            for j in (0...totalColumns) {
                if(board[i][j] == "P") {
                    board[i][j] = "O"
                }
            }
        }
    }
}

```