Given a 2-dimensional grid of integers, each value in the grid represents the color of the grid square at that location.

Two squares belong to the same *connected component* if and only if they have the same color and are next to each other in any of the 4 directions.

The *border* of a connected component is all the squares in the connected component that are either 4-directionally adjacent to a square not in the component, or on the boundary of the grid (the first or last row or column).

Given a square at location (r0, c0) in the grid and a color, color the border of the connected component of that square with the given color, and return the final grid.

**Example 1:**

**Input:** grid = [[1,1],[1,2]], r0 = 0, c0 = 0, color = 3

**Output:** [[3, 3], [3, 2]]

**Example 2:**

**Input:** grid = [[1,2,2],[2,3,2]], r0 = 0, c0 = 1, color = 3

**Output:** [[1, 3, 3], [2, 3, 3]]

**Example 3:**

**Input:** grid = [[1,1,1],[1,1,1],[1,1,1]], r0 = 1, c0 = 1, color = 2

**Output:** [[2, 2, 2], [2, 1, 2], [2, 2, 2]]

**Note:**

1. 1 <= grid.length <= 50
2. 1 <= grid[0].length <= 50
3. 1 <= grid[i][j] <= 1000
4. 0 <= r0 < grid.length
5. 0 <= c0 < grid[0].length
6. 1 <= color <= 1000