Spring 2024

CS 412 (Algorithms: Design and Analysis)

Weekly Challenge 04:

Announced: Friday, February 9, 2024.
Deadline: Friday, February 16, 2024 (11:59 pm PKT).
Total marks: 1.

Instructions: Submit **individually** your solution as a PDF with the file name as your *studentID.pdf*; typeset in LaTeX. You must submit your solution on Canvas.

---

1. (1 point) Imagine that you have a 2D grid where each cell contains a color value (see Figure 1 for an example). Given a start location (x, y), target color, and replacement color, design a recursive algorithm to fill closed regions with a specific color (replacement color). Clearly define a base criterion and analyze the worst-case time complexity of your proposed algorithm/approach.

   Suppose that you have the following array:

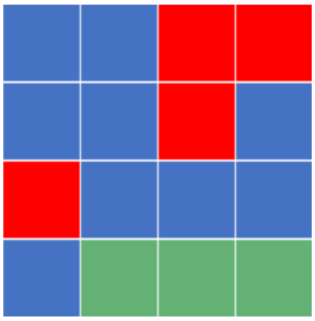

Figure 1: Before

Now, we're given a *start_position* $= (0, 0)$ and *new_color* $=$ orange. As we can see the color of cell $(0, 0)$ is blue. Therefore, we'll recolor all its connected cells that have the same color to the *new_color*:

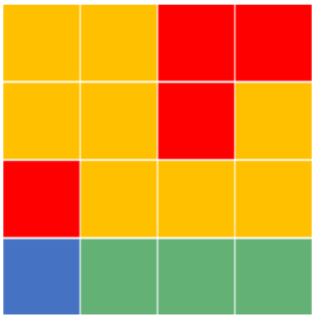

Figure 2: After

Input: start location (row, column), replacement color.
Output: replaces all adjacent cells with the target color with the replacement color. The target color is the original color of the starting location.
Moves: possible moves are up, down, left, right. In addition to up, down, left, and right, you can also add up-right, up-left, down-right, and down-left.
Hint: The algorithm might include recursive calls for each move.

Note: Figure 1 is only an example, your algorithm should be valid for any number of colors and any arrangement of the colors.

Solution:

Algorithm 1 Fill Region

```
function FillRegion(grid, start_row, start_col, target_color, replacement_color)
    if $\neg(0 \leq \text{start\_row AND start\_row} < \text{len(grid)})$ AND $(0 \leq \text{start\_col AND start\_col} < \text{len(grid[0])})$ then
        return
    end if
    if $\text{grid}[start\_row][start\_col] \neq \text{target\_color}$ then
        return
    end if
    if $\text{grid}[start\_row][start\_col] = \text{replacement\_color}$ then        ▷ Already visited
        return
    end if
    $\text{grid}[start\_row][start\_col] \leftarrow \text{replacement\_color}$        ▷ Mark as visited
    FillRegion(grid, start_row + 1, start_col, target_color, replacement_color)        ▷ Explore adjacent cells
    FillRegion(grid, start_row − 1, start_col, target_color, replacement_color)
    FillRegion(grid, start_row, start_col + 1, target_color, replacement_color)
    FillRegion(grid, start_row, start_col − 1, target_color, replacement_color)
end function
```