Rotate By 90 Degree

solved by	Senan
Platform	GeeksForGeeks
difficulty	Medium
tags	Matrix Swap
language	C++
solved on	@05/11/2024
link	https://www.geeksforgeeks.org/problems/rotate-by-90-degree0356/1
Completion	☑

Intuition

Rotating a matrix 90 degrees clockwise can be visualized as flipping the matrix upside down (mirroring it vertically) and then transposing it. This simplifies the problem into two basic operations: a vertical flip followed by a transpose.

Approach

1. **Mirror the Matrix Vertically**: Swap rows from top to bottom (e.g., the first row with the last, the second with the second-to-last, etc.).
2. **Transpose the Matrix**: Swap elements such that `matrix[i][j]` becomes `matrix[j][i]` for all `i < j`. This operation turns rows into columns and completes the 90-degree rotation.

Complexity

Time Complexity:

- **O(n^2)**: Both mirroring and transposing require visiting each element of the matrix once.

Space Complexity:

- **O(1)**: The rotation is done in-place, so no additional space proportional to the input size is used.

Code

```
void rotate(vector<vector<int> >& matrix) {
    int n = matrix.size();
    // Mirror the matrix vertically
    for(int i = 0; i < n / 2; i++) {
        swap(matrix[i], matrix[n - i - 1]);
    }
    // Transpose the matrix
    for(int i = 0; i < n; i++) {
        for(int j = i + 1; j < n; j++) {
            swap(matrix[i][j], matrix[j][i]);
        }
```

```
    }
}
```