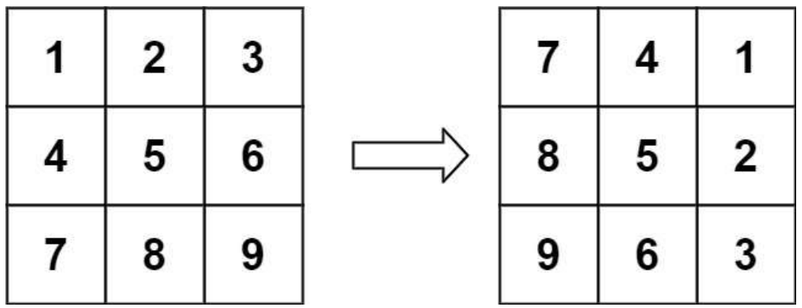
48. Rotate Image

Medium 8443 465 Add to List Share

You are given an `n x n` 2D `matrix` representing an image, rotate the image by **90** degrees (clockwise).

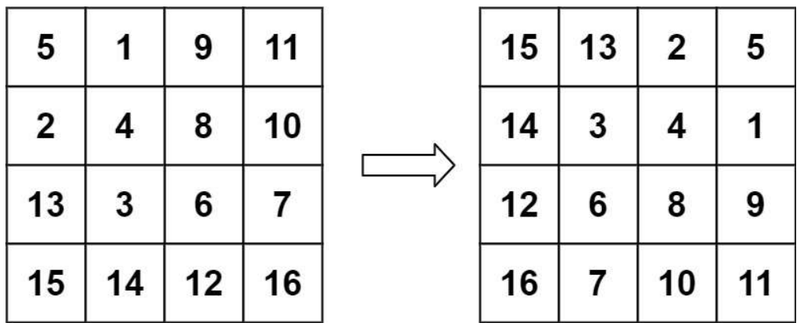
You have to rotate the image **in-place**, which means you have to modify the input 2D matrix directly. **DO NOT** allocate another 2D matrix and do the rotation.

Example 1:




```
Input: matrix = [[1,2,3],[4,5,6],[7,8,9]]
Output: [[7,4,1],[8,5,2],[9,6,3]]
```

Example 2:




```
Input: matrix = [[5,1,9,11],[2,4,8,10],[13,3,6,7],[15,14,12,16]]
Output: [[15,13,2,5],[14,3,4,1],[12,6,8,9],[16,7,10,11]]
```

Java Autocomplete

```java
class Solution {
    public void rotate(int[][] matrix) {

        int dimension = matrix.length;

        //Step 1 :- Transpose the matrix
        for(int i = 0; i < dimension; i++){
            for(int j = 0; j < i + 1; j++){
                int temp = matrix[i][j];
                matrix[i][j] = matrix[j][i];
                matrix[j][i] = temp;
            }
        }

        //Step 2 :- Reverse the rows
        for(int i = 0; i < dimension; i++){
            for(int j = 0; j < dimension / 2; j++){
                int temp = matrix[i][j];
                matrix[i][j] = matrix[i][dimension - j - 1];
                matrix[i][dimension - j - 1] = temp;
            }
        }
    }
}
```

Testcase Run Code Result Debugger

Accepted Runtime: 0 ms

Your input `[[1,2,3],[4,5,6],[7,8,9]]`

Output `[[7,4,1],[8,5,2],[9,6,3]]` Diff

Expected `[[7,4,1],[8,5,2],[9,6,3]]`