Rotate Image

Description

Given an n x n 2D matrix representing an image, rotate the image by 90 degrees (clockwise). Rotation should be performed in-place, which means modifying the input 2D matrix directly without allocating another matrix.

Examples

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]]

Constraints

- n == matrix.length == matrix[i].length
- 1 <= n <= 20
- -1000 <= matrix[i][j] <= 1000

Approach

The rotation can be achieved layer by layer, starting from the outermost layer and moving towards the inner layers. Within each layer, each element is moved in a cyclic manner.

Usage

```
# Test the solution

solution = Solution()

matrix1 = [[1,2,3],[4,5,6],[7,8,9]]

solution.rotate(matrix1)

print(matrix1)  # Output: [[7,4,1],[8,5,2],[9,6,3]]


matrix2 = [[5,1,9,11],[2,4,8,10],[13,3,6,7],[15,14,12,16]]

solution.rotate(matrix2)

print(matrix2)  # Output: [[15,13,2,5],[14,3,4,1],[12,6,8,9],[16,7,10,11]]
```