

**Solution 1:**

class Solution:

    def rotate(self, matrix: List[List[int]]) -> None:

        """

        Do not return anything, modify matrix in-place instead.

        """

        l, r = 0 ,len(matrix)-1

        while l< r:

            for i in range(r-l):

                t, b = l, r

                # save the top left

                topLeft = matrix[t][l+i]

                # move bottom left into top left

                matrix[t][l+i] = matrix[b-i][l]

                # move bottom right into bottom left

                matrix[b-i][l] = matrix[b][r-i]

                # move top right into bottom right

                matrix[b][r-i] = matrix[t+i][r]

                # move top left into top right

                matrix[t+i][r] = topLeft

            r -=1

            l +=1

        #print(matrix)

run = Solution()

run.rotate([[1,2,3],[4,5,6],[7,8,9]])

**Output:**

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

**Time Complexity:** O(n2) - as here many repeated work is doing

**Space Complexity:** O(1) – using input memory