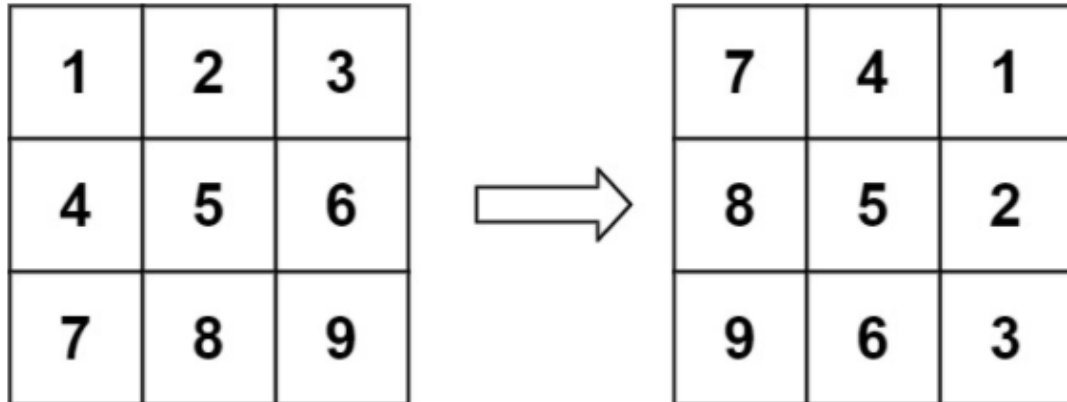


旋转图像

给定一个 $n \times n$ 的二维矩阵 `matrix` 表示一个图像。请你将图像顺时针旋转 90 度。

你必须在 [原地](#) 旋转图像，这意味着你需要直接修改输入的二维矩阵。请不要使用另一个矩阵来旋转图像。

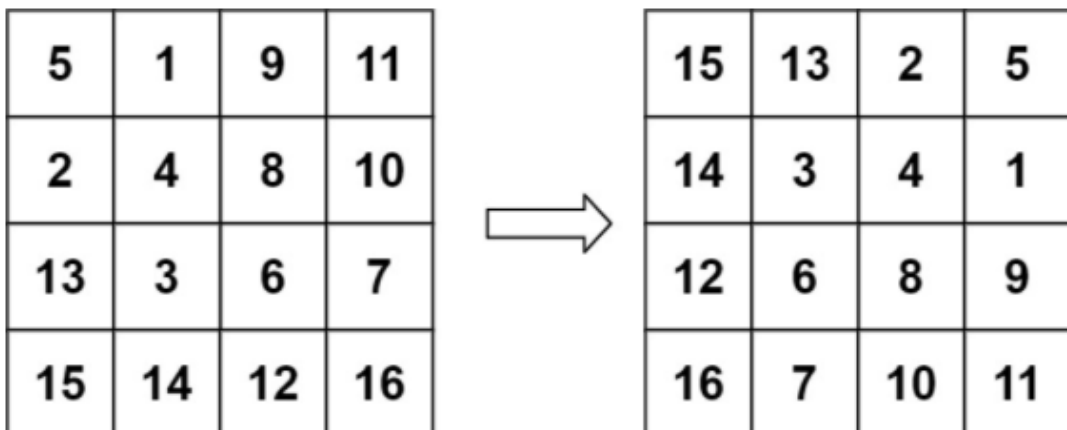
示例 1:



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

输出: `[[7,4,1],[8,5,2],[9,6,3]]`

示例 2:



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

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

示例 3:

输入: `matrix = [[1]]`

输出: `[[1]]`

示例 4:

输入: matrix = [[1,2],[3,4]]

输出: [[3,1],[4,2]]

```
class Solution {
public:
    void rotate(vector<vector<int>>& matrix) {
        int n=matrix.size();
        //第一步: 上下翻转
        for(int i=0;i<n/2;i++)
        {
            for(int j=0;j<n;j++)
                swap(matrix[i][j],matrix[n-i-1][j]);
        }
        //第二步: 左上/右下对角线翻转
        for(int i=0;i<n;i++)
        {
            for(int j=0;j<i;j++)
                swap(matrix[i][j],matrix[j][i]);
        }
    }
};
```