Description | Editorial | Solutions (10.8K) | Submissions

## 48. Rotate Image

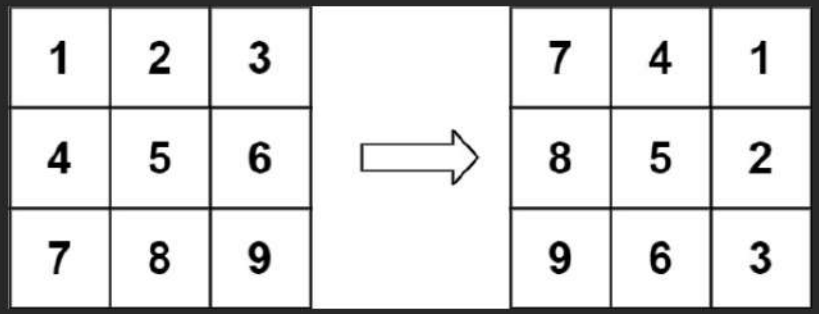
Medium 15.8K 692

Companies

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

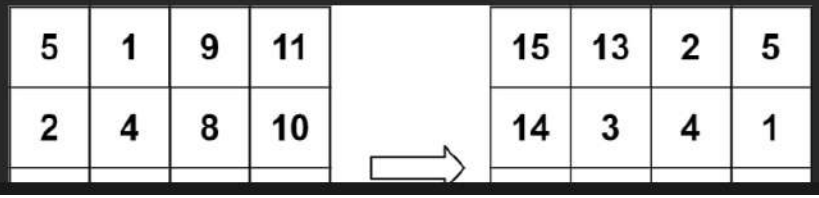
You have to rotate the image in-place, which means you have to modify the input 2D matrix directly. **DO NOT** allocate another 2D matrix and do the rotation.

**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:**




Java | Auto

```java
class Solution {
    public void rotate(int[][] matrix) {
        int n = matrix.length;

        //Column Reverse
        for(int i = 0; i < n; i++){
            int a = 0;
            int b = n-1;
            while(a <= b){
                int temp = matrix[a][i];
                matrix[a][i] = matrix[b][i];
                matrix[b][i] = temp;
                a++;
                b--;
            }
```

Ln 29, Col 2

Testcase | Result

Accepted Runtime: 0 ms

• Case 1 • Case 2

Input

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

Output

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

Expected

Console Run Submit

Description | Editorial | Solutions (330) | Submissions

# 1489. Find Critical and Pseudo-Critical Edges in Minimum Spanning Tree

Hint

Hard | 1.3K | 109

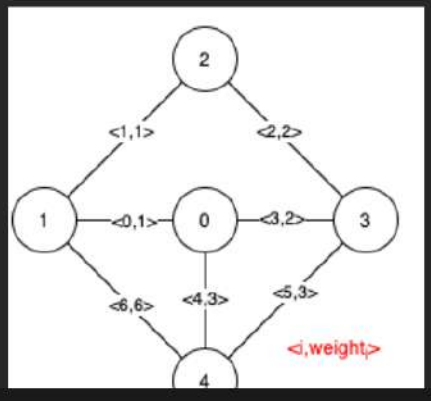
Companies

Given a weighted undirected connected graph with `n` vertices numbered from `0` to `n - 1`, and an array `edges` where `edges[i] = [a_i, b_i, weight_i]` represents a bidirectional and weighted edge between nodes `a_i` and `b_i`. A minimum spanning tree (MST) is a subset of the graph's edges that connects all vertices without cycles and with the minimum possible total edge weight.

Find *all the critical and pseudo-critical edges in the given graph's minimum spanning tree (MST)*. An MST edge whose deletion from the graph would cause the MST weight to increase is called a *critical edge*. On the other hand, a pseudo-critical edge is that which can appear in some MSTs but not all.

Note that you can return the indices of the edges in any order.

**Example 1:**




Java | Auto

```java
import java.util.*;

class UnionFind {
    private int[] parent;

    public UnionFind(int n) {
        parent = new int[n];
        for (int i = 0; i < n; i++)
            parent[i] = i;
    }

    public int findParent(int p) {
        return parent[p] == p ? p : (parent[p] = findParent(parent[p]));
    }

    public void union(int u, int v) {
```

Ln 78, Col 2

Testcase | Result

Accepted Runtime: 1 ms

• Case 1 • Case 2

Input

n =
5

edges =
[[0,1,1],[1,2,1],[2,3,2],[0,3,2],[0,4,3],[3,4,3],[1,4,6]]

Output

[[0,1],[2,3,4,5]]

Console | Run | Submit