Lucky Number - LC : 1380

CODE:

import java.util.ArrayList;

import java.util.List;

public class LuckyNumber {

    static List<Integer> solution(int arr[][]) {

        List<Integer> minRow = new ArrayList<>();

        List<Integer> maxCol = new ArrayList<>();

        List<Integer> result = new ArrayList<>();

        int minRowE = Integer.MAX\_VALUE;

        int maxColE = Integer.MIN\_VALUE;

        for(int i = 0; i < arr.length; i++) {

            for(int j = 0; j < arr[i].length; j++) {

                minRowE = Math.min(minRowE, arr[i][j]);

            }

            minRow.add(minRowE);

            minRowE = Integer.MAX\_VALUE;

        }

        for(int i = 0; i < arr.length; i++) {

            for(int j = 0; j < arr[i].length; j++) {

                maxColE = Math.max(maxColE, arr[j][i]);

            }

            maxCol.add(maxColE);

            maxColE = Integer.MIN\_VALUE;

        }

        for(int each : minRow) {

            if(maxCol.contains(each)) {

                result.add(each);

            }

        }

        return result;

    }

    public static void main(String[] args) {

        int matrix[][] = {{3,7,8},{9,11,13},{15,16,17}};

        List<Integer> ans = solution(matrix);

        System.out.println(ans);

    }

}

Rotate Image - LC – 48

CODE:

class Solution {

    public void transpose(int[][] matrix){

        for(int i = 0; i < matrix.length; i++){

            for(int j = i; j < matrix[0].length; j++){

                int temp = matrix[i][j];

                matrix[i][j] = matrix[j][i];

                matrix[j][i] = temp;

            }

        }

    }

    public void reverse(int[] arr){

        int i = 0, j = arr.length-1;

        while(i <= j){

            int temp = arr[i];

            arr[i] = arr[j];

            arr[j] = temp;

            i++;

            j--;

        }

    }

    public void rotate(int[][] matrix) {

        transpose(matrix);

        for(int[] arr : matrix)

            reverse(arr);

    }

}

OUTPUT :

A screenshot of a computer

Description automatically generated with medium confidence

Spiral Matrix - LC : 54

CODE:

public class spiralmat {

    public static void main(String[] args) {

        int arr[][] = {

            {1,2,3,5},

            {4,6,7,1},

            {1,5,8,2},

            {5,6,8,9}

        };

        // number of rows

        int m = arr.length;

        // number of cols

        int n = arr[0].length;

        // start row index

        int k = 0;

        // start col index

        int l = 0;

        int i;

        while(k < m && l < n) {

            // print rows (first row)

            for(i = l; i < n; i++) {

                System.out.print(arr[k][i] + ",");

            }

            k++;

            // print cols (last col)

            for(i = k; i < m; i++) {

                System.out.print(arr[i][n-1] + ",");

            }

            n--;

            // print last row

            if(k < m) {

                for(i = n - 1; i >= l; i--){

                    System.out.print(arr[m-1][i] + ",");

                }

                m--;

            }

            if(l < n) {

                for(i = m - 1; i >= k; i--) {

                    System.out.print(arr[i][l] + ",");

                }

                l++;

            }

        }

    }

}

OUTPUT :

