//1. Write an efficient algorithm to search a value in a 2D matrix in which the rows are in sorted order and the first integer of each row is greater than the last integer of the previous row.

function findX(mat,x)

{

    row=mat.length;

    column=mat[0].length;

    i=0;

    j=column-1;

    while(i<row&& j>=0)

    {

        if(mat[i][j]==x)

        return ' Found';

        else if(mat[i][j]<x)

        i++;

        else if(mat[i][j]>x)

        j--;

    }

    return ' Not found';

}

let arr =

[[2,14,15,16],

 [7,18,20,22],

 [8,21,23,24],

 [10,26,27,28]

];

let n=8;

console.log(findX(arr,n));

//2. Find the row index which has maximum no. of unique elements in a matrix efficiently.

const mat=

[[2,14,5,18],

[10,8,1,22],

[8,21,22,15],

[10,14,21,28]]

//output-2,8

function unique(mat) {

    let map = {};

    for(let row of mat) {

        for(let num of row) {

            map[num] = (map[num] || 0) + 1;

        }

    }

    for(let key in map) {

    if(map[key] == 1) {

        console.log(key);

    }

}

}

unique(mat);