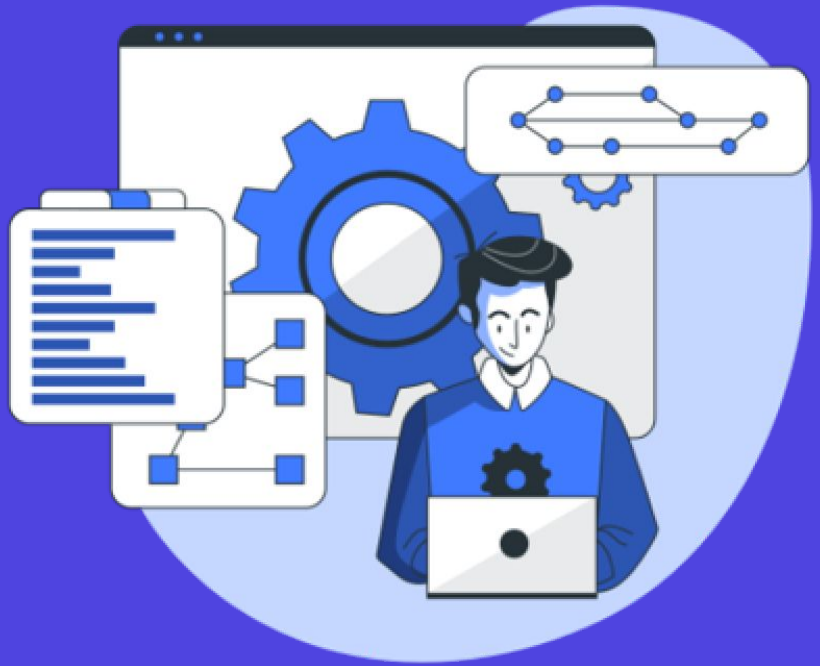
2D Arrays-Part-2



Relevel
by Unacademy

Problem - 1

Given a 2D array A, return a boolean value indicating that the given matrix is a diagonal matrix or not. A matrix is said to be diagonal if all the elements of the array other then the diagonal is 0
If there will be zero in the diagonal then also, we will print output as true if all remaining cells have zero. Having zeros in diagonal does not affect our output

Input:

A=[[1,0,0,0],
[0,2,0,0],
[0,0,3,0],
[0,0,0,4]]

Output:

true

Explanation:

Since all the elements of the array other then the diagonal element are 0, it can be considered as a diagonal matrix

Solution-

Code Link - https://jsfiddle.net/4juazeph/

JavaScript + No-Library (pure JS) ▾

```
//Logic function to verify the cells other then diagonal
function checkMatrixDiagonal(A)
    {
        //Iterate through the 2D Array
        for (let i = 0; i < A.length; i++)
            for (let j = 0; j < A.length; j++)
                if ((i != j) &&  (A[i][j] != 0)) // Check if current cell is not diagonal one and its value is not 0 - if yes return false
                    return false;

        return true;
    }

//Taking Input
let A=[    [0,1,0,0],
  [0,0,0,0],
  [0,0,0,0],
  [0,0,0,0] ]


//Execute the logic
console.log(checkMatrixDiagonal(A));
```

Problem - 2

Given a 2D array of size MxN, you need to display N integers which denotes the column wise addition of the 2D array

Input:

M=4,N=3

3	4	5
3	4	2
2	3	4
4	4	4

Output:

12,15,15

Explanation:

Sum of column-1 : 3+3+2+4= 12
Sum of column-2 : 4+4+3+4 = 15
Sum of column-3 : 2+3+4+4 = 15

Solution-

Code Link - https://jsfiddle.net/2mgetfps/

JavaScript + No-Library (pure JS) ▼

```
//M = Rows, N = Columns
let M=4,N=3
//A = 2DArray
const A = [[3,4,5],[3,4,2],[2,3,4],[4,4,4]]
let col_sum=[]

//Iterate through  columns first
for(let idx = 0;idx<N;idx++){
  let sum_=0
  //Iterate through  rows
  for(let idx2 = 0;idx2<M;idx2++){
    sum_+=A[idx2][idx];  //Add values from the column cells
  }
  col_sum.push(sum_);
}

//print the solution
console.log(col_sum);  //12,15,15
```

Approach - 2

Intuition:
We can reduce it by applying column wise addition on two rows at a time. I.e.

3	4	5
3	4	2

On using reduce, we will apply same logic with the next row
[6,8,7]
+[2,3,4]
—--------
[8,11,11]
Again using reduce we add this with next row
[8,11,11]
+[4,4,4]
—----------
[12,15,15]

Solution - 2

Code Link - https://jsfiddle.net/yapc5v2d/

JavaScript + No-Library (pure JS) ▾

```
//M = Rows, N = Columns
let M=4,N=3

//A = 2DArray
const A = [[3,4,5],[3,4,2],[2,3,4],[4,4,4]]


var col_sum = A.reduce((A, B) => A.map((X, idx) => X + B[idx])); // Logic to fetch the sum

//print the solution
console.log(col_sum); //12,15,15
```

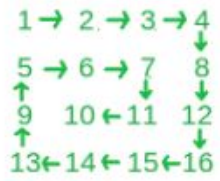
Problem - 3

Given a 2D array. Our task is to print a 2D array in the spiral form

2D Array -

```
1   2   3   4
5   6   7   8
9   10  11  12
13  14  15  16
```

Spiral Form -



Output - `1 2 3 4 5 6 12 18 17 16 15 14 13 7 8 9 10 11`

Approach

Based on the spiral form, it is clear that we can solve this problem using loops and conditionals. If we follow the above image of spiral form, we can see that there are 4 directions where we need to move in order to print the output. We can follow those 4 directions to get our desired output.

Let's have a look step by step -

1. Initialize 4 variables - e(start row index), f(end row index), g(start column index), h(end column index)
2. Create a loop and iterate over it until all elements got printed
3. For each outer loop, print elements in clockwise direction
4. Print eth row(from column index g to h) and increase k -> this will print top row
5. Print (h - 1)th column(from row index e to f) and decrease h -> this will print right column
6. Print (f - 1)th row (from column index (h - 1) to g) and decrease f -> this will print bottom row
7. Print (g)th row(from row index f-1 to e) and increase g -> this will print left column

Solution

Code link - https://jsfiddle.net/0nhf6q79/

Problem - 4

Given a 2D array where the value of any row element is greater than the previous row. Our task is to find if element x is present in the array.

Input:

2	14	15	16
7	18	20	22
8	21	23	24
10	26	27	28

X= 21

Output:
True

Explanation:
Position of element 15 in the array is 2,1, so it exists hence true

Approach

In the problem statement, it is mentioned that every row element is greater than the previous row element. So we can start from the top right element and can move either left or downward based upon the comparison of the value.

Steps -

1. Initialize i and j to top right element of matrix i.e. -> i=0, j=col-1
2. Iterate through the matrix and check if current element is our target element - if yes print output
3. If current element < target element -> increment the row
4. If current element > target element -> decrement the column

Solution

Code Link - https://jsfiddle.net/wh8jureg/

JavaScript + No-Library (pure JS)

```
function search(A,x){
        if (A.length == 0)
            return false
        let row = A.length
        let col = A[0].length
        let i = 0;
        let j = col-1;
        while (i < row && j >= 0){
            if (A[i][j] == x) // If current element is our target element -> print output
                return true
            else if (A[i][j] < x) // If current element < target element -> increment the row
                i += 1
            else if (A[i][j] > x) // If current element > target element -> decrement the column
                j -= 1
        }
        return false
}


 let x = 6
A=[[2,4,5,6],[7,8,10,12],[13,15,16,18],[20,21,22,23]]
console.log(search(A,x))
```

Problem - 5

Given a 2D array. Our task is to find the maximum element of each row.

Input:

2	14	15	16
7	18	20	22
8	21	23	24
10	26	27	28

Output:

16
22
24
28

Approach

In the problem statement, it is mentioned that we need to find the maximum from each row. We can iterate each row and find the maximum element in that particular row.

Steps -

1. Initialize output array with size = number of rows
2. Iterate through row and find the maximum
3. Save maximum element in output array for respective row
4. Print output array

Solution

Code link - https://jsfiddle.net/r9xeavLg/

Problem - 6

Given a 2D array. Our task is to find the unique elements. A unique element is an element whose frequency is 1 i.e. it is not repeating in the whole 2D Array.
If there is no any unique element, print message as “No unique element found”

Input:

2	14	15	18
10	18	14	22
8	21	22	15
10	14	21	28

Output:

2
8

Approach

In the problem statement, it is mentioned that we need to find the unique element. We can use a count array which is storing count of elements. If count = 1, we will print those elements in output

Steps -

1. FInd maximum element of the given array as max
2. Initialize 1D array having size = max
3. Iterate through input array and increment value in count array for index = element
4. Iterate through count array and print index if value is equal to 1

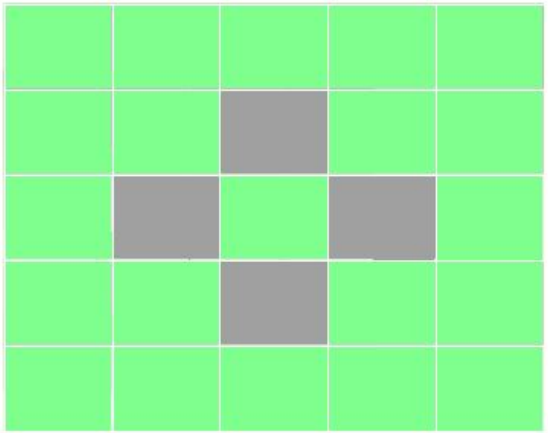
Solution

Code link - https://jsfiddle.net/j38uhwve/

Practice Question

Problem-1:
Given a 1D array of size 10 , convert it into a 2D array of size 2x5

Problem-2:
Given a 2D array, find the sum of the diagonal and the boundary elements of it.
In the given matrix of size 5x5 the colored cell marks the diagonal and the boundary elements



Input:
A=[[1,2,3,4,1],
[5,6,7,8,2],
[9,10,11,12,13],
[13,14,15,16,15],
[11,12,15,19,15],
]

Output
195

Thank You!