Matrix Spiral

Given an (MxN) matrix of integers, return an array in spiral order.  
  
Input: Array of integers  
Output: Array of integers

# Example

Input: [[1,2,3], => Output: [1, 2, 3, 6, 9, 8, 7, 4, 5]  
 [4,5,6],  
 [7,8,9]]

Input: [[]] => Output: []

# Constraints

Time Complexity: O(MN)  
Auxiliary Space Complexity: O(MN)  
  
  
Values of the array will be digits 0-9.

# Solution

1. Create a max and min for the X and Y coordinates.
2. While X min <= X max and Y min < Y max, perform 4 loops adding values into a results array
   1. Loop through the top row and increment Y min.
   2. Loop through the last column and decrement X max
   3. Loop through the bottom row and then decrement Y max
   4. Loop through the first column and then increment Y
3. Once the while loop breaks, return the result array.

# Notes

Must account long or tall matrices:  
Input: [[1,2,3,4]] => Output: [1, 2, 3, 4]

Note: 2c and 2d need checks to ensure values are pushed in more than once

Javascript Solution

function Matrix Spiral(matrix) {

if(!matrix.length) {return []; }

var yMin = 0;

var xMin = 0;

var yMax = matrix.length - 1;

var xMax = matrix[0].length - 1;

var results = [];

while (xMin <= xMax && yMin <= yMax) {

for(var i = xMin; i <= xMax; i++){

results.push(matrix[yMin][i]);

}

yMin++;

for(i = yMin; i <= yMax; i++){

results.push(matrix[i][xMax]);

}

xMax--;

if(yMin <= yMax){

for(i = xMax; i >= xMin; i--){

results.push(matrix[yMax][i]);

}

yMax--;

}

if(xMin <= xMax){

for(i = yMax; i >= yMin; i--){

results.push(matrix[i][xMin]);

}

xMin++;

}

}

return results;

};

TripleByte has asked this problem on their final round.

# Resources

https://leetcode.com/problems/spiral-matrix/