We have a 2D array filled with positive integers. The goal here is to start from the top left position and finish at the bottom right and return the minimum cost of the paths taken. So, I must transverse through the 2D array scanning the integers in the elements adding up the smallest ones.

What must be done fist is creating a method to take in the given matrix and return and int.

I will check if the grid is null, or if there is anything within the elements. If nothing, return 0

There create 2 for loops, one to initialize the top row and the other for the left column.

A nested for-loop will be used to loop through the array, scanning for the minimum values.

Return the array

class Solution {

public int minFallingPathSum(int[][] a) {

if (a == null || a.length == 0) {

return 0;

}

int d = a.length;

int s = a[0].length;

for (int i = 1; i < a.length; i++) {

a[i][0] = a[i - 1][0] + a[i][0];

}

for (int j = 1; j < a[0].length; j++) {

a[0][j] = a[0][j - 1] + a[0][j];

}

for (int i = 1; i < a.length; i++) {

for (int j = 1; j < a[0].length; j++) {

a[i][j] = Math.min(a[i - 1][j], a[i][j - 1]) + a[i][j];

}

}

return a[d - 1][s - 1];

}

}