1605. Find Valid Matrix Given Row and Column Sums

Solved

Medium | Topics | Companies | Hint

You are given two arrays `rowSum` and `colSum` of non-negative integers where `rowSum[i]` is the sum of the elements in the i^{th} row and `colSum[j]` is the sum of the elements of the j^{th} column of a 2D matrix. In other words, you do not know the elements of the matrix, but you do know the sums of each row and column.

Find any matrix of **non-negative** integers of size `rowSum.length x colSum.length` that satisfies the `rowSum` and `colSum` requirements.

Return *a 2D array representing **any** matrix that fulfills the requirements*. It's guaranteed that **at least one** matrix that fulfills the requirements exists.

Example 1:

```
Input: rowSum = [3,8], colSum = [4,7]
Output: [[3,0],
         [1,7]]
```

```
class Solution {
    func restoreMatrix(_ rowSum: [Int], _ colSum: [Int]) -> [[Int]]
{
        var mat =
Array(repeating:Array(repeating:0,count:colSum.count),count:rowSum
.count)

        for i in (0..<rowSum.count) {
            mat[i][0] = rowSum[i]
        }

        for i in (0..<colSum.count-1) {
            var sum = 0
            var ex = colSum[i]
            for j in (0..<rowSum.count) {
                sum = sum + mat[j][i]
            }
            if(sum == ex) {
                continue
            } else {
                var needtoreduce = sum - ex
                for j in (0..<rowSum.count) where needtoreduce > 0
{
                    var t = min(needtoreduce,mat[j][i])
                    mat[j][i] = mat[j][i] - t
                    mat[j][i+1] = t
                    needtoreduce = needtoreduce - t
                }
```

```
            }
        }

        return mat
    }
}
```