Problem #1582: Special Posiitons in a Binary Martrix

<https://leetcode.com/problems/special-positions-in-a-binary-matrix/description/>

My Solution:

Solution #1:

Brute Force method:

1. Initialize count to 0.
2. Let m be the length of mat (number of rows) and n be the length of the first row

(number of columns).

1. Check for cell value to be 1 in the matrix mat.

If so, then check if the row total is 1.

If do, check if column total called col\_total is 1.

If so, increment count by 1.

class Solution:

def numSpecial(self, mat: List[List[int]]) -> int:

count = 0

m = len(mat) # number of rows

n = len(mat[0]) # number of columns

for i in range(m):

for j in range(n):

if mat[i][j] == 1:

if sum(mat[i]) == 1:

col\_total = 0

for k in range(m):

col\_total += mat[k][j]

if col\_total == 1:

count += 1

return count

Solution #2: A Better Solution.

Only if the row total is 1, do we check if the column total of the cell having 1 is 1 as well.

class Solution:

def numSpecial(self, mat: List[List[int]]) -> int:

count = 0

m = len(mat) # number of rows

n = len(mat[0]) # number of columns

for i in range(m):

if sum(mat[i]) == 1:

for j in range(n):

if mat[i][j] == 1:

col\_total = 0

for k in range(m):

col\_total += mat[k][j]

if col\_total == 1:

count += 1

return count