##when you sum up the path(I,j), what if the obstacle is on (I,j) exactly?

class Solution(object):

def uniquePathsWithObstacles(self, obstacleGrid):

"""

:type obstacleGrid: List[List[int]]

:rtype: int

"""

if not obstacleGrid:

return 0

m,n = len(obstacleGrid),len(obstacleGrid[0])

if m==1:

for i in range(n):

if obstacleGrid[0][i]:

return 0

return 1

if n==1:

for i in range(m):

if obstacleGrid[i][0]:

return 0

return 1

dp = [[0 for i in range(n)] for i in range(m)]

dp[0][0] = 1

ind=0

for i in range(n):

if obstacleGrid[0][i]:

dp[0][i] = 0

ind = i

break

dp[0][i] = 1

if ind>0:

for i in range(ind+1,n):

dp[0][i] = 0

ind = 0

for i in range(m):

if obstacleGrid[i][0]:

dp[i][0] = 0

ind = i

break

dp[i][0] = 1

if ind>0:

for i in range(ind+1,m):

dp[i][0] = 0

for i in range(1,m):

for j in range(1,n):

a,b = 0,0

if not obstacleGrid[i-1][j]:

a = dp[i-1][j]

if not obstacleGrid[i][j-1]:

b = dp[i][j-1]

dp[i][j] = a+b

return dp[-1][-1] if not obstacleGrid[-1][-1] else 0