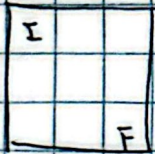


63. Unique Paths II



→ abajo, der

→ input vector <vector<int>> obstacleGrid

→ usar dp en imaginario

→ dp → matriz que contabiliza la cantidad de maneras que se llega a grid[i][j]

Grid

0	0	0
0	1	0
0	0	0

dp

1	0	0
0	0	0
0	0	0

→ comienza de arriba, solo hay una manera de llegar a la primera posición, así que solo se pone 1

dp

1	1	0
1	0	0
0	0	0

dp

1	1	1
1		0
1	0	0

→ al moverse a la derecha, el cuadrado solo tiene una manera de como llegar a este. De igual forma para abajo.

1	1	1
1	0	1
1	1	0

1	1	1
1	0	1
1	1	2

→ al llegar a la esquina, se chequea la cantidad de caminos que se puede llegar hasta la posición de arriba y la izquierda. La suma de estos lleva la cantidad total de caminos posibles.

caso $grid[i][j] == 1$, $dp[i][j] = 0$

caso contrario

if revisar izq

if revisar arriba