solving method : dynamic programming

Let M[i][j] be the maximum sum achievable among all paths from (1,1) to (i,j)

initialization:

M[1][1]=X[1][1];

M[i][1]=M[i-1][1]+X[i][1];

M[1][i]=M[1][i-1]+X[1][i];

Optimal Substructure:

M[i][j]= X[i][j]+max {M[i-1][j], M[i][j-1]}

That is, there are two ways to go to (i,j) : from (i-1,j) or (i,j-1). If there is an optimal way from (1,1) to (i,j) and it passes by (i-1,j) , it must go from (1,1) to (i-1,j) first in an optimal way, and then go to (i,j).

Optimal solution= M[n][n].

Complexity: