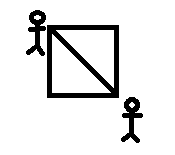
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **National University of Computer and Emerging Sciences, Lahore Campus** | | | | |
| C:\Users\saif\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.Word\final design.jpg | **Course:** | **Design & Analysis of Algorithms** | **Course Code:** | **CS2009** |
| **Program:** | **BS (Computer Science)** | **Semester:** | **Spring 2023** |
| **Duration:** | **15 Minutes** | **Total Marks:** | **12** |
| **Paper Date:** | **30-March-2023** | **Weight:** | **4** |
| **Section:** | **H** | **Page(s):** | **1** |
| **Exam:** | **Quiz 3** | **Reg. No.** |  |
| **Instruction/Notes:** |  | | | |

**Question 1: CLO 1, [12 marks]**

You are stranded on a square island with your friend, but, you are at one end of the island and your friend is at the opposite end of the diagonal connecting you two as shown below:



You want to go to your friend. However, the island is floored with square marble tiles and stepping on each tile has a certain cost. You task is to go to your friend in the minimum cost possible. You can move in any direction but not jump and skip tiles. Write a bottom up dynamic programming algorithm that takes no more than O(n2) if there are n2 tiles in total.

**Solution:**

A[1..N][1..M]

A[1][1] = 0

FOR (i = 2 to N)

A[i][1] =A[i-1][1] + W[i-1][1].down

FOR (j=2 to M)

A[1][j] = A[1][j-1] + W[1][j-1].right

FOR(I= 2 to N)

FOR (j = 2 to M)

A[i][j] = min(A[i-1][j] + W[i-1][j].down, A[i][j-1]+W[i][j-1].right,A[i-1][j-1]+W[i-1][j-1].diagonal)

Final answer is stored in A[N][M]

OR

A[1..N][1..M]

A[1][1] = 0

FOR (i = 2 to N)

A[i][1] = A[i-1][1] + W[i][1]

FOR (j = 2 to M)

A[1][j] = A[1][j-1] + W[1][j]

FOR (i = 2 to N)

FOR (j = 2 to M)

A[i][j] = min(A[i-1][j] + W[i][j], A[i][j-1] + W[i][j], A[i-1][j-1] + W[i][j])

Final answer is stored in A[N][M]