Quiz 1

Solution:

1) We use dynamic programming to solve this problem.

Let's say we have i=1,2,…,k stage and in each stage, we have j=1,2,…,m nodes. In order to find the shortest path from source t to a node (i,j), we need to take into consideration the shortest weight of node (i-1,j). Assuming $M_{i,j}$ store the shortest weight of node (i,j), then it can be calculated using the following recursive formula: $M_{i,j} = min_{1\le p\le m}\{M_{i-1,p} + W_{i-1,p,i,j}\}$

2) Pseudocode:

Input: k (# of stage), m (# of node), $M_{i,j}$ (store the minimum weight at node i, j)

Output: shortest path M_{opt} from s to t

```
Set M_{i,j} = ∞
For t=1 to m
        M_{1,t} = W_{s,t}
For i from 2 to k do
        For j from 1 to m do
                For p from 1 to m do
                        M_{i,j} = min{M_{i,j}, M_{i-1,p} + W_{i-1,p,i,j}}
                End
        End
End
M_{opt} = min_{1≤p≤m}{M_{k,p} + W_{k,p,t}}
```

---

3) Prove:

In this program, $M_{i,j}$ store the shortest weight of from s to node (i,j). At stage 1, the weight of all the nodes are initialized with the weight from s to node (i=1,j). Then starting from stage 2, the shortest path from s to node is represented by the recursive part: $M_{i,j} = min_{1\le p\le m}\{M_{i-1,p} + W_{i-1,p,i,j}\}$.When the shortest path of node (i,j) is calculated, all the

$M_{i-1,j}$ will be considered and only the one with minimum weight be the optimal solution. We keep running this recursive function until we reach stage k.

At the end, we loop over the $M_{k,j}$ once together with the $W_{k,j,t}$ to find the shortest path from s to t. Therefore, this will always give the optimal solution

4) Here we have three loops so time complexity is $O(km^2)$, k represents the number of stage and m represents the number of nodes at each stage