Mubarra Khan Z5161249 Friday, 31 July 2020

Q4

Given a weighted directed graph G(V, E), find a path in G (possibly self intersecting) of length exactly K that has the maximum total weight. The path can visit a vertex multiple times and can traverse an edge also multiple times. It can also start and end at arbitrary vertices or even start and end at the same vertex.

Solution:

This problem will use the Floyd-Warshall algorithm which finds the shortest path with minimum cost from any vertex to any other vertex.

So for this problem we will need to modify the algorithm in such a way that it will find the path with maximum weight with exactly K length.

The algorithm :

max_weight (w)

- Let n be the number of vertices the graph.
- $D^0 = w$
    - This is because initially the maximum weight of the matrix is the weight provide
- For values from 1 to k - at this step we ensure the path is of exactly length K
    - Let $D^k = d_{i,j}^k$ as D holds the maximum path weight
    - For all values from i from 1 to n (as n is the number of vertices)
        - For all values j from 1 to n
            - We will set $d_{i,j}^k$ to the maximum value between $d_{i,j}^{k-1}$ and $d_{ik}^{k-1} + d_{kj}^{k-1}$
            - The above step is a modified step to the algorithm to find the maximum weighted path
- At the end we will return D^k

Analysis :

We updated matrix D at every iteration so we can update the max weighted path between every pair of vertices.

At last when the algorithm returns the matrix D^k, that is the maximum weight path between every vertex with path length exactly K. It then chose the maximum valued element in that matrix.

Time Complexity : O(KE)