Recitation Problems - Com S 311

Week of Apr 15^{th} - Apr 20^{th}

1. Given a weighted directed graph with non-negative weights, write an algorithm in pseudocode to find the shortest path distances **to** a specific vertex (say t) from all other vertices.

   Reverse the edges of the graph. Apply Dijkstra's algorithm with source vertex as t. The shortest path distances from t to all vertices in the reverse graph is equal to the shortest path distances from all vertices to t in the original graph.

Algorithm 1 (given a graph $G = (V, E, wt)$ and a vertex t)

// Construct new graph $G' = (V', E')$ with reversed edges
Initialize a new graph G' with vertices $V' = V$
for each $(v, u) \in E$ **do**
    add a directed edge from u to v in G' with weight of $wt(v, u)$
$Dijkstra(G', t)$
return the d-values of each of the vertices in V'

2. Given a weighted directed graph with non-negative weights and three vertices u, v and w, write an algorithm in pseudocode to verify that w is present in some shortest path from u to v.

   Apply Dijkstra's algorithm from u and from w. The first run will compute the shortest path distances from u to w and from u to v. The second run will compute the shortest path distance from w to v. If the shortest path distance from u to w and from w to v is equal to shortest path distance from u to v, then return true; otherwise return false.

Algorithm 2 (given a graph $G = (V, E, wt)$ and vertices u, v, w)

Initialize a copy of G and call it G' where $G' = (V, E, wt)$
$Dijkstra(G, u)$ (suppose the d-values computed here are stored in $d(v)$ for all $v \in V$)
$Dijkstra(G', w)$ (suppose the d-values computed here are stored in $d'(v)$ for all $v \in V$)
if $d(w) + d'(v) == d(v)$ **then**
    return $true$
return $false$

3. Prove the following claim: for any cycle in a graph, the edge with the highest weight in the cycle does not belong to a MST.

   Proof by contradiction.

   Assume: $e = (u, v)$ be a highest weight edge in a cycle and it belongs to some MST T.

   Not all edges in the cycle belongs to T (as T is a tree). Now, put the vertices reachable from u (without considering edge e) in graph in group A and put the vertices reachable from v (without considering e) in graph in group B. We have a cut of the graph (partition of vertices of the graph). Look at the cross-edges: e is a cross-edge. There should be some other cross-edges as well because e is part of a cycle. Furthermore, the edge e is not the min-wt cross-edge.

Therefore, it can be exchanged for a cross-edge that has smaller weight and create a new spanning tree that has a weight smaller than T.

Therefore, T is not an MST and our assumption is not valid.