Algorithms I

Tutorial 7 Solution Hints

October 28, 2016

Problem 1

Let $v_1, v_2, \ldots v_n$ be a topological sorting of G where $n = |V|$. Let $dp[v_i]$ denote the number of paths from s to v_i. Let $v_x = u$.

$$dp[v_i] = \begin{cases} 0, & i < x \\ 1, & i = x \\ \sum_{(v_j,v_i)\in E(G)} dp[v_j], & otherwise \end{cases}$$

By definition, $dp[t]$ gives the number of paths from s to t

Problem 2

First we show that G is a tree when both BFS-tree and DFS-tree are same. If G and T are not same, then there should exist an edge $e(u, v)$ in G, that does not belong to T. In such a case:

- in the DFS-tree, one of the u or v, should be an ancestor of the other.
- in the BFS-tree, u and v can differ by only one level.

Since, both DFS-tree and BFS-tree are same tree T , it follows that one of u and v should be an ancestor of the other and they can differ by only one level. This implies that the edge connecting them must be in T . So, there can not be any edges in G which are not in T.

In the second part of the proof: Since G is a tree, each node has a unique path from the root. So, both BFS and DFS produce the same tree, and the tree is same as G.

Problem 3

We start BFS from u. Since each edge has cost 1, the length of shortest path between u and v is the level of u in the BFS-tree. If v is not in the same component as u, there is no path between them.

Similarly, if all the edges have cost c, the length of the shortest path between u and v will be $level_v \cdot c$.

Problem 4

Here are the steps to find the center:

1. if G has at most 2 vertices, output $V(G)$ as the center.
2. Remove all leaves (vertices with degree 1) from G and corresponding edges
3. Go to 1

Problem 5

Suppose initially the minimum spanning tree was T. After each edge weight is increased by 1, the minimum spanning tree changes to T'. Therefore there will be at least an edge $(u, v) \in E(T)$ but $(u, v) \notin T'$. Suppose we add edge (u, v) to the tree T'. $T^* = T' + (u, v)$. Since (u, v) was not in T' therefore (u, v) must be the longest edge in the cycle C formed in T^* . But since (u, v) is the longest edge it cannot be in the MST T'. (u, v) is the longest edge and therefore when we decrease each edge weight by 1, (u, v) will still be the longest edge in cycle C formed in T. But longest edge (u, v) can not be contained in MST T . Therefore $(u, v) \notin T$ which is a contradiction. It implies that trees T and T' are same.

Problem 6

1. $e \notin E'$ and $w'(e) > w(e)$: T is still MST
2. $e \notin E'$ and $w'(e) < w(e)$: We add this edge to T. Now we've got exactly 1 cycle. Based on cycle property in MST, we need to find and remove edge with highest value on that cycle. We can do it using DFS or BFS.
3. $e \in E'$ and $w'(e) < w(e)$: T is still MST
4. $e \in E'$ and $w'(e) > w(e)$: We remove this edge from T. Now we have 2 connected components that should be connected. We can find both components using BFS or DFS. Now, we need to find an edge with smallest value that connects these components which we can find by iterating over the edges.