CS180 Spring 2013

Homework 3

The following homework is due Thursday, April 25 at the beginning of lecture.

When submitting your homework, please include your name at the top of each page. If you submit multiple pages, please staple them together. We also ask you to indicate which name is your last name on the first page, such as underlining it.

Please provide complete arguments and time complexity analysis for all solutions

1. DEPTH-FIRST SEARCH TREE. Given a graph G, starting from node s, we traverse the graph through depth-first search (DFS). In the depth-first search algorithm, consider the moment when you follow some edge leading out of node u to reach node v that is first discovered by the DFS algorithm. At this moment, we add the edge (u, v) to the set of edges E. Prove that the set of edges in E comprises a tree. We call the tree that is produced in this way a depth-first search tree.

2. Given a graph $G = (V, E)$ and a tree $T = (V, E')$, T is a spanning tree of G, that is, the set of nodes in T is the same as the set of nodes in G, and the set of edges E' in T is a subset of the set of edges E in G. Design an algorithm to determine if there exists a depth-first search in graph G to create a *depth-first search tree* that is the same as T. (You could start a depth-first search from any node v in the graph)

3. Design an algorithm to find the diameter of a tree by using breadth-first search twice.

4. Given a DAG G, each edge in G has a weight. Given two nodes s and t in the graph, design an algorithm to find the weighted longest path from node s to node t.