Homework Assignment #8

Note

This assignment is due 2:10PM Monday, May 30, 2011. Please write or type your answers on A4 (or similar size) paper. Drop your homework by the due time in Yih-Kuen Tsay's mail box on the first floor of Management College Building II. Late submission will be penalized by 20% for each working day overdue. You may discuss the problems with others, but copying answers is strictly forbidden.

There are five problems in this assignment, each accounting for 20 points.

Problems

1. (7.1) Consider the problem of finding balance factors in binary trees discussed in class (see slides for "Design by Induction"). Solve this problem using DFS. You need only to define preWORK and postWORK.

2. (7.3) Given as input a connected undirected graph G, a spanning tree T of G, and a vertex v, design an algorithm to determine whether T is a valid DFS tree of G rooted at v. In other words, determine whether T can be the output of DFS under some order of the edges starting with v. The running time of the algorithm should be $O(|E| + |V|)$.

3. (7.28) A **binary de Bruijn sequence** is a (cyclic) sequence of 2^n bits $a_1 a_2 \cdots a_{2^n}$ such that each binary string s of size n is represented somewhere in the sequence; that is, there exists a unique index i such that $s = a_i a_{i+1} \cdots a_{i+n-1}$ (where the indices are taken modulo 2^n). For example, the sequence 11010001 is a binary de Bruijn sequence for $n = 3$. Let $G_n = (V, E)$ be a directed graph defined as follows. The vertex set V corresponds to the set of all binary strings of size $n-1$ ($|V| = 2^{n-1}$). A vertex corresponding to the string $a_1 a_2 \cdots a_{n-1}$ has an edge leading to a vertex corresponding to the string $b_1 b_2 \cdots b_{n-1}$ if and only if $a_2 a_3 \cdots a_{n-1} = b_1 b_2 \cdots b_{n-2}$. Prove that G_n is a directed Eulerian graph, and discuss the implications for de Bruijn sequences.

4. In the topological sorting algorithm that we discussed in class for directed acyclic graphs, DFS is used to calculate the indegree of each vertex in the input graph. Please give a detailed description of this calculation in an adequate pseudo code.

5. (7.38) Given a directed acyclic graph $G = (V, E)$, find a simple (directed) path in G that has the maximum number of edges among all simple paths in G. The algorithm should run in linear time.