CS 170, Spring 1997
Midterm #1

Problem #1

(50 points) Let T be a binary tree with root r. The height of a vertex v is 0 if v is a leaf vertex, and otherwise it is the height of the subtree rooted at v; i.e. it is the distance between v and the farthest leaf in the subtree rooted at v. The balance factor of a vertex v is defined to be the difference between the height of v's left child and the height of its right child. By convention, the balance factor of leaf vertex is defined to be 0. Give a linear time algorithm that given as input a binary tree T and its root vertex r, computes the balance factor of all vertices in T.

Problem #2 (50 points)

You are given a directed graph G(V,E) with associated weights w:E->Z on the edges. Moreover you know that there is only one edge (u, v) whose weight is negative -- all other edges weight are positive.

(30 points) Show how to use Dijkstra's algorithm to determine whether G(V,E) has negative cycle. What is the running time of your algorithm?

(20 points) Assume that G(V,E) does not have a negative cycle. Given a vertex s in V, show how to use Dijkstra's algorithm to find shortest paths from s to every vertex in G(V,E). Justify the correctness of your algorithm, and analyze its running time.

Problem #3

(50 points) You are given an undirected graph G(V,E) with associated edge weights w:E->Z+. You are also given an acyclic subset F in E of edges. Give an efficient algorithm to find a spanning tree T of G(V,E) such that F in T and such that T has the minimum weight among all spanning trees of G(V,E) that contain F. Justify the correctness of your algorithm, and analyze its running time.

(Note: 20 points for the algorithm, 20 points for correctness proof, 10 points for running time).
