Math 245, May 12, 2010
Test 3

This is a group test. You are strongly encouraged to work in a group, though not required. If you do work in a group, you may turn in one test per group. Use a separate sheet of paper and write the names of all the group members on top. Make sure to explain your answers to the best of your ability. There are six problems on the test, worth a total of 100 points.

A *tree* is a connected graph containing no circuits.

All the questions below pertain to trees.

---

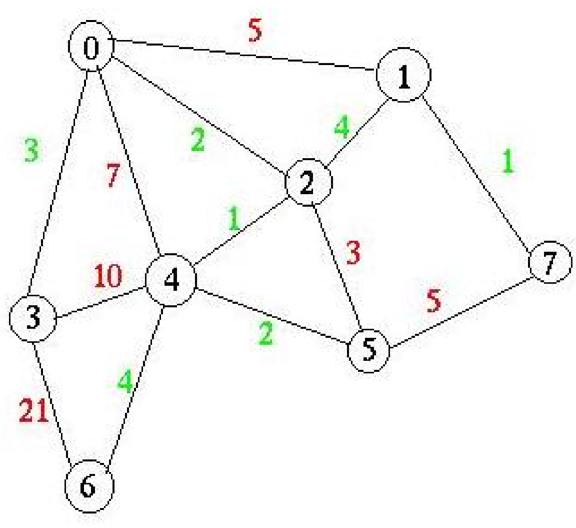
1. (10 points) Give at least two different examples of situations where trees can be used.

2. (15 points) Give an informal proof that there is exactly one path between any two vertices of a tree.

3. (20 points) Show that a tree on p vertices has $p-1$ edges. **Hint:** You will prove this by strong induction on the number of vertices. Assume that the claim holds for every tree on less than p vertices. Consider a tree on p vertices and remove an edge from it. You can use the fact that if you remove an edge from a tree, the resulting graph is either a union of a tree and a vertex or of two trees.

4. (15 points) If G is a graph on p vertices, and the number of edges is $p-1$, must G be a tree? Explain.

5. (20 points) Suppose you want to build a railroad that connects a certain number of cities. You want to construct the system as cheaply as possible, even if this will inconvenience the passengers. How should the railroad be built? Note that there will be no circuits in your railroad (Why?) and that it will have to be connected. This means that you are looking for the cheapest tree that spans all the cities. Such a tree is called a *minimal spanning tree* (MST) of a graph.



There are two algorithms that find a minimal spanning tree: Kruskal's and Prim's.

Kruskal's Algorithm:

Kruskal's algorithm works as follows: Take a graph with n vertices, and keep adding the shortest (least cost) edge, while avoiding the creation of cycles, until $(n-1)$ edges have been added. (NOTE: Sometimes two or more edges may have the same cost. The order in which the edges are chosen, in this case, does not matter. Different MSTs may result, but they will all have the same total cost, which will always be the minimum cost.)

Prim's Algorithm:

This algorithm builds the MST one vertex at a time. It starts at any vertex in a graph (vertex A, for example), and finds the least cost vertex (vertex B, for example) connected to the start vertex. Now, from either A or B, it will find the next least costly vertex connection, without creating a cycle (vertex C, for example). Now, from either A, B, or C, it will find the next least costly vertex connection, without creating a cycle, and so on. Eventually, all the vertices will be connected, without any cycles, and an MST will be the result. (NOTE: Two or more edges may have the same cost, so when there is a choice by two or more vertices that is exactly the same, then one will be chosen, and an MST will still result.)

Use both Kruskal' and Prim's algorithms to find the minimum spanning tree. Did you get the same result?

6. (20 points) Agents A, B, C, D, E, F, G, and H are political conspirators in what has become known as the "Blottergate" affair. In order to coordinate their cover-up efforts, it is vital that each agent be able to communicate directly or indirectly with every other conspirator. Such communications involve a certain amount of risk for everyone. Below is a table of "risk factors" associated with direct communication between the indicated parties. All other direct communications are too likely to expose the cover-up scheme. What is the least total risk involved in a connecting system?

Agent pairs	A B	A C	A E	A F	A G	B C	B F	C D	C F	C G	C H	D E	D H	E H
Risk factor	9	3	8	3	4	10	6	6	4	5	7	6	3	5