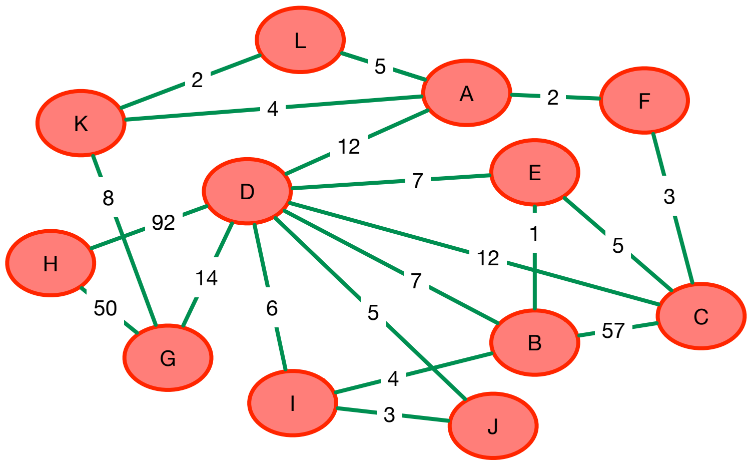
Greedy Algorithms
Due: Monday, October 30

Directions: Some of the questions on this assignment will appear on the quiz on Monday, October 30.

1. Find the minimum spanning tree for the following graph.



   (a) What is the weight of the MST?
   (b) How many MSTs are there?

2. Let $G = (V, E)$ be an undirected graph. Prove that if all edge weights are distinct, then it has a unique minimum spanning tree.

3. Design and analyze an algorithm to find the *maximum spanning tree*, that is, a spanning tree with largest total weight.

4. Prove: *If e is any edge of minimum weight in G, then e must be a part of some minimum spanning tree.*

5. Prove or give a counterexample: *If G is a graph on more than $|V| - 1$ edges, and there is a unique heaviest edge, then this edge cannot be a part of any minimum spanning tree.*

6. Design and analyze an algorithm that takes a tree and determines if it has a *perfect matching*: a set of edges that touches each vertex exactly once.

7. Alice wants to throw a party and is deciding whom to invite. She has n people to choose from, and she has made up a list of which pairs of these people know each other. She wants to pick as many people as possible, subject to two constraints: at the party, each person should have at least five other people whom they know and five other people whom they don't know. Give an efficient algorithm that takes as input the list of n people and the list of pairs who know each other and outputs the best choice of party invitees. Give the running time in terms of n.

8. Suppose you are given n ropes of different lengths, you need to connect these ropes into one rope. The cost to connect two ropes is equal to sum of their lengths. Design and analyze a greedy algorithm to connect the ropes with minimum cost.

   For example: Suppose you have three ropes with lengths 2, 5, and 8. If you chose first to connect the length 5 and 8 ropes, then connect the length 2 and 13 ropes, the total cost would be $(5 + 8) + (13$

+ 2) = 28. However, if you first chose to connect the length 2 and 5 ropes, then the length 7 and 8 ropes, the total cost would be (2 + 5) + (7 + 8) = 22 (which happens to be optimal).

9. Fill in the pseudocode for the construction of the Huffman tree in the Huffman Coding algorithm. It is marked with a star on the slides.

10. Suppose that symbols a, b, c, d, e occur with frequencies $1/2, 1/4, 1/8, 1/16, 1/16$ respectively.

    (a) What is a Huffman encoding of the alphabet?

    (b) If this encoding is applied to a file consisting of 1,000,000 characters with the given frequencies, what is the length of the encoded file in bits?

11. We use Huffman's algorithm to obtain an encoding of the alphabet a, b, c with frequencies f_a, f_b, f_c. In each of the following cases, either give an example of frequencies that would yield the specific code, or explain why the code cannot possibly be obtained (no matter what the frequencies are).

    (a) Code: $0, 10, 11$

    (b) Code: $0, 1, 00$

    (c) Code: $10, 01, 00$

12. Suppose we are allowed values 0, 1, or 2 (instead of just 0 or 1) for constructing an optimal prefix-free code. Create a modified Huffman algorithm for compressing sequences of characters from an alphabet of size n, where the characters occur with known frequencies $f_1, f_2, \ldots, f_n$. Your algorithm should encode each character with a variable-length codeword over the values 0, 1, 2 such that no codeword is a prefix of another codeword and so as to obtain the maximum possible compression. Prove your algorithm generates a code that is prefix-free.