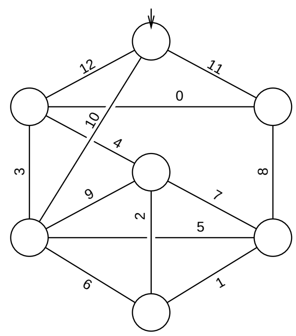
1. Using any method you like, compute the following subgraphs for the weighted graph below. Each subproblem is worth 3 points. Each incorrect edge costs you 1 point, but you cannot get a negative score for any subproblem.

   (a) a depth-first search tree, starting at the top vertex;

   (b) a breadth-first search tree, starting at the top vertex;

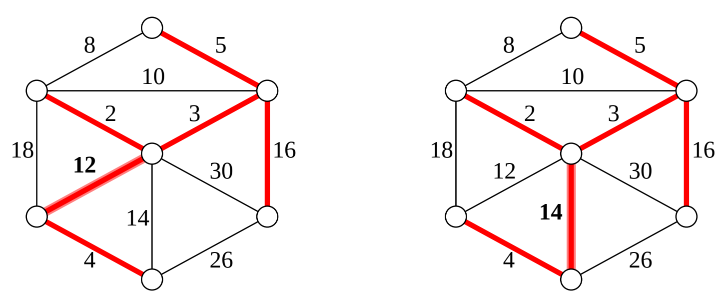
   (c) a shortest path tree, starting at the top vertex;

   (d) the minimum spanning tree.



2. Suppose you are given a weighted undirected graph G (represented as an adjacency list) and its minimum spanning tree T (which you already know how to compute). Describe and analyze and algorithm to find the *second-minimum spanning tree* of G, *i.e.*, the spanning tree of G with smallest total weight except for T.

   The minimum spanning tree and the second-minimum spanning tree differ by exactly one edge. But *which* edge is different, and *how* is it different? That's what your algorithm has to figure out!



The minimum spanning tree and the second-minimum spanning tree of a graph.

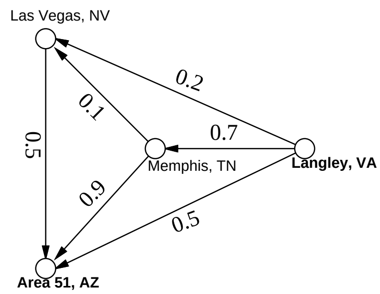
3. (a) *[4 pts]* Prove that a connected acyclic graph with V vertices has exactly $V - 1$ edges. ("It's a tree!" is not a proof.)

   (b) *[4 pts]* Describe and analyze an algorithm that determines whether a given graph is a tree, where the graph is represented by an adjacency list.

   (c) *[2 pts]* What is the running time of your algorithm from part (b) if the graph is represented by an adjacency matrix?

4. Mulder and Scully have computed, for every road in the United States, the exact probability that someone driving on that road *won't* be abducted by aliens. Agent Mulder needs to drive from Langley, Virginia to Area 51, Nevada. What route should he take so that he has the least chance of being abducted?

   More formally, you are given a directed graph $G = (V, E)$, where every edge e has an independent safety probability $p(e)$. The *safety* of a path is the product of the safety probabilities of its edges. Design and analyze an algorithm to determine the safest path from a given start vertex s to a given target vertex t.



   With the probabilities shown above, if Mulder tries to drive directly from Langley to Area 51, he has a 50% chance of getting there without being abducted. If he stops in Memphis, he has a $0.7 \times 0.9 = 63\%$ chance of arriving safely. If he stops first in Memphis and then in Las Vegas, he has a $1 - 0.7 \times 0.1 \times 0.5 = 96.5\%$ chance of being abducted![1]

5. *[1-unit grad students must answer this question.]*

   Many string matching applications allow the following *wild card* characters in the pattern.

   - The wild card `?` represents an arbitrary single character. For example, the pattern `s?r?ng` matches the strings `string`, `sprung`, and `sarong`.
   - The wild card `*` represents an arbitrary string of zero or more characters. For example, the pattern `te*st*` matches the strings `test`, `tensest`, and `technostructuralism`.

   Both wild cards can occur in a single pattern. For example, the pattern `f*a??` matches the strings `face`, `football`, and `flippityfloppitydingdongdang`. On the other hand, neither wild card can occur in the text.

   Describe how to modify the Knuth-Morris-Pratt algorithm to support patterns with these wild cards, and analyze the modified algorithm. Your algorithm should find the first substring in the text that matches the pattern. An algorithm that supports only one of the two wild cards is worth 5 points.

[1] That's how they got Elvis, you know.