Tasks and HW 2

Task 1. Draw binary trees to represent the expressions below. Are they unique?
1. $a * b - (c/(d + e))$ 2. $a/(b - c * d)$ 3. $a + b + c + d$ 4. $a + b * c - d = e$

Task 2. Graph G is defined via the following adjacency list:
• 1: 2; • 2: 1, 4, 5, 6; • 3: 4; • 4: 2, 3; • 5: 2, 6, 7, 8; • 6: 2, 5; • 7: 5, 9; • 8: 5, 10; • 9: 7, 10; • 10: 8, 9, 11, 12, 13, 14; • 11: 10; • 12: 10; • 13: 10, 15; • 14: 10, 16; • 15: 13, 16; • 16: 14, 15.

1. Draw graph G
2. Suppose two random edges are removed from G. Calculate the expected value and the variance of the number of components of the resulting graph.

Task 3. Find all possible spanning trees for each of the graphs in figures 1a and 1b.
Task 4. Find a spanning tree for each of the graphs in figures 1c and 1d.
Task 5. Use Kruskal's algorithm to find a minimum spanning tree for each of the graphs in figures 1e and 1f. Indicate the order in which edges are added to form each tree.
Task 6. Use Prim's algorithm starting with vertex a or v_0 to find a minimum spanning tree for each of the graphs in figures 1e and 1f. Indicate the order in which edges are added to form each tree.
Task 7. For each of the graphs in figures 1g and 1h, find all minimum spanning trees that can be obtained using (a) Kruskal's algorithm and (b) Prim's algorithm starting with vertex a or t. Indicate the order in which edges are added to form each tree
Task 8. A pipeline is to be built that will link six cities. The cost (in hundreds of millions of dollars) of constructing each potential link depends on distance and terrain and is shown in the weighted graph from firgure 1i. Find a system of pipelines to connect all the cities and yet minimize the total cost.
Task 9. Use Dijkstra's algorithm to find the shortest path from a to z for each of the graphs. In each case make tables representing the action of the algorithm. (a) figure 1j; (b) figure 1k; (c) figure 1g with $a = a$ and $z = f$; (d) figure 1h with $a = u$ and $z = w$.
Task 10. Prove that any two spanning trees for a graph have the same number of edges.
Task* 11. Prove that given any two distinct vertices of a tree, there exists a unique path from one to the other.

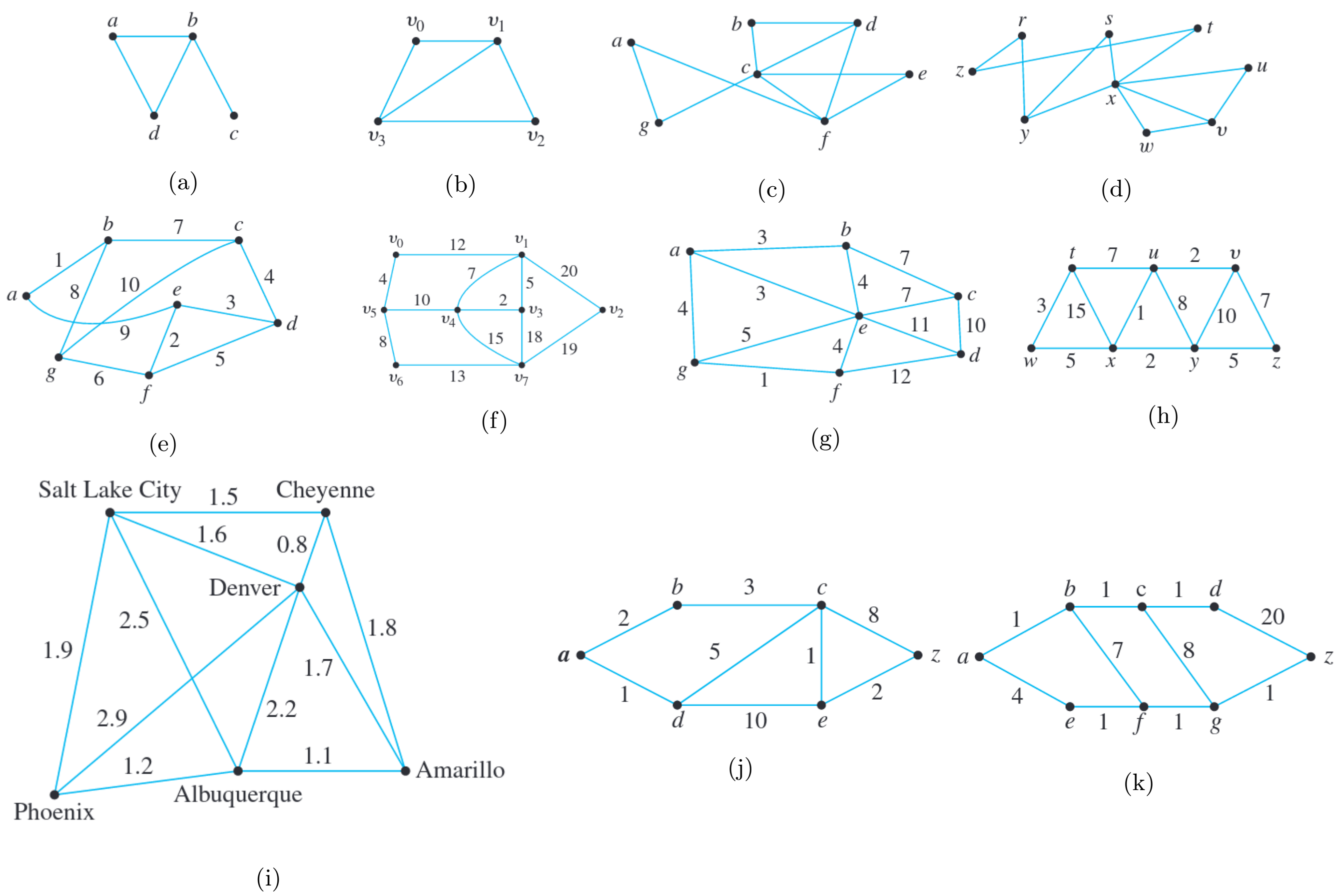

Figure 1: graphs

Task* 12. *Bottleneck spanning tree* — a spanning tree in which the most expensive edge is as cheap as possible. Come up with an algorithm building (minimum) bottleneck spanning tree. What is its compexity?
Task* 13.

1. Suppose T_1 and T_2 are two different spanning trees for a graph G. Must T_1 and T_2 have an edge in common? Prove or give a counterexample.

2. Suppose that the graph G in part 1 is simple. Must T_1 and T_2 have an edge in common? Prove or give a counterexample.

Task* 14. Prove that an edge e is contained in every spanning tree for a connected graph G if, and only if, removal of e disconnects G.
Task* 15. If G is a connected, weighted graph and no two edges of G have the same weight, does there exist a unique minimum spanning tree for G?
Task* 16. Prove that if G is a connected, weighted graph and e is an edge of G that (1) has greater weight than any other edge of G and (2) is in a circuit of G, then there is no minimum spanning tree T for G such that e is in T.
Task* 17. Suppose a disconnected graph is input to Kruskal's algorithm. What will be the output?
Task* 18. Suppose a disconnected graph is input to Prim's algorithm. What will be the output?
Task 19 [Programming]. Implement • Kruscal's • Prim's • Dijkstra's algorithms.
Task 20 [Programming]. Solve at least one of the following tasks.

1. Implement Prim's algorithm such that its complexity be at most $O(|E| * \log |V|)$.

2. Implement algorithm building (minimum) bottleneck spanning tree.

3. By a given graph find all critical edges in minimum spanning tree.