Homework Goals
• Develop knowledge of Trees and Tree algorithms;
• Develop knowledge of Binary Trees and properties;
• Develop knowledge of Binary Search Trees (in comparison to heaps);
• Develop knowledge of Graphs and Graph algorithms.

Trees: 10/50 points
1. Text, Problem C-6.28, Page 308. For simplicity, assume T is a binary tree. Also assume that you are given a function depth(Tree, node) that returns depth of node in Tree.

   Algorithm LCA(T,v,w)
   Input: tree T, node v, and node w
   Output: node common

2. The following questions refer to Figure 6.2, page 255 in the text.
   a) List the nodes, given a preorder traversal of the tree.

   b) List the nodes, given a postorder traversal of the tree.

   c) List the nodes, given a level order traversal of the tree.

   d) The tree in Figure 6.2 is a general tree. Convert the general tree into a binary tree as shown in class. Draw the converted binary tree below. List the nodes, given an inorder traversal of the converted binary tree.
**Binary Trees: 10/50 points**

3. Draw a proper binary tree with a *height* of 4 and a maximum number of leaf nodes.

*Note that for the questions below (4a – 4d), the binary tree is not necessarily proper.*

4a) What is the minimum number of *leaf* nodes in a binary tree with height $h$?

4b) What is the maximum number of *leaf* nodes in a binary tree with height $h$?

4c) What is the minimum number of nodes in a binary tree with height $h$?

4d) What is the maximum number of nodes in a binary tree with height $h$?

**Min Heaps and Binary Search Trees: 10/50 points**

5. Briefly describe (1-2 sentences) the binary search tree property.

6. Briefly describe (1-2 sentences) the **min**-heap property.
7. Draw a binary search tree containing the following keys: 12, 2, 5, 9, 11, 7, 1. Assume that the keys are inserted into the binary search tree in the order given.

8. Draw a min-heap as a tree containing the following keys: 12, 2, 5, 9, 11, 7, 1. Assume that the keys are inserted into the heap in the order given.

9. Can the binary search tree property be used to print the keys of an n-node binary search tree in sorted order in O(n) time? If so, describe the algorithm. If not, explain why not.

10. Can the min-heap property be used to print the keys of an n-node min-heap in sorted order in O(n) time? If so, give the algorithm. If not, explain why not.
**Graphs: 10/50 points**

In a *directed* graph, the *out-degree* of a vertex is the number of edges leaving it, and the *in-degree* of a vertex is the number of edges entering it.

11. Given an adjacency *matrix* representation of a directed graph of $V$ vertices and $E$ edges, how long does it take to compute the out-degree of all vertices in terms of big-oh? Briefly explain.

**Complexity**

12. Given an adjacency *matrix* representation of a directed graph of $V$ vertices and $E$ edges, how long does it take to compute the in-degree of all vertices in terms of big-oh? Briefly explain.

**Complexity**

13. Given an adjacency *list* representation of a directed graph of $V$ vertices and $E$ edges, how long does it take to compute the out-degree of all vertices in terms of big-oh? Briefly explain.

**Complexity**

14. Given an adjacency *list* representation of a directed graph of $V$ vertices and $E$ edges, how long does it take to compute the in-degree of all vertices in terms of big-oh? Briefly explain.

**Complexity**
Graph Algorithms: 10/50 points
You are given the graph shown below:

15. Show each iteration of Prim’s algorithm on the graph with node B as the starting node. The first iteration is provided for you.

<table>
<thead>
<tr>
<th>Iteration 1</th>
<th>Iteration 2</th>
<th>Iteration 3</th>
<th>Iteration 4</th>
<th>Iteration 5</th>
<th>Iteration 6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>k</td>
<td>d</td>
<td>p</td>
<td>k</td>
<td>d</td>
</tr>
<tr>
<td>A</td>
<td>F</td>
<td>3</td>
<td>B</td>
<td>A</td>
<td></td>
</tr>
<tr>
<td>B</td>
<td>T</td>
<td>0</td>
<td>-</td>
<td>B</td>
<td></td>
</tr>
<tr>
<td>C</td>
<td>F</td>
<td>6</td>
<td>B</td>
<td>C</td>
<td></td>
</tr>
<tr>
<td>D</td>
<td>F</td>
<td>-</td>
<td>-</td>
<td>D</td>
<td></td>
</tr>
<tr>
<td>E</td>
<td>F</td>
<td>8</td>
<td>B</td>
<td>E</td>
<td></td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>-</td>
<td>-</td>
<td>F</td>
<td></td>
</tr>
</tbody>
</table>

16. Draw the resulting minimal spanning tree. Include edge weights in your drawing.