CMSC 341 Data Structures

General Tree Review

These questions will help test your understanding of the general tree material discussed in class and in the text. These questions are only a study guide. Questions found here may be on your exam, although perhaps in a different format. Questions NOT found here may also be on your exam.

General Trees

1. Define *tree.*

2. Define *k-ary tree.*

3. For any tree, T, define the following

   a. path in T
   b. length of a path in T
   c. height of a node in T
   d. depth of a node in T
   e. height of T
   f. depth of T
   g. external node
   h. internal node
   i. leaf

4. Given the drawing of an arbitrary tree, draw the first-child, next-sibling representation of the tree.

5. Given the first-child, next-sibling representation of a tree, draw the tree.

6. Prove that there are *n* – 1 edges in any tree with *n* nodes.

7. What is the worst-case Big-Oh performance for the **insert, find** and **remove** operations in a general tree? Why is this so?

8. Write a recursive member function of the “static K-ary” tree class that counts the number of nodes in the tree.

Binary Trees

1. Define *binary tree*, *full binary tree, complete binary tree* and *perfect binary tree*

2. Prove that a perfect binary tree of height h has 2^h leaf nodes.

3. Prove that a perfect binary tree of height h has $2^{h+1} - 1$ nodes.

4. Prove that a full binary tree with n internal nodes has $n + 1$ leaf nodes.

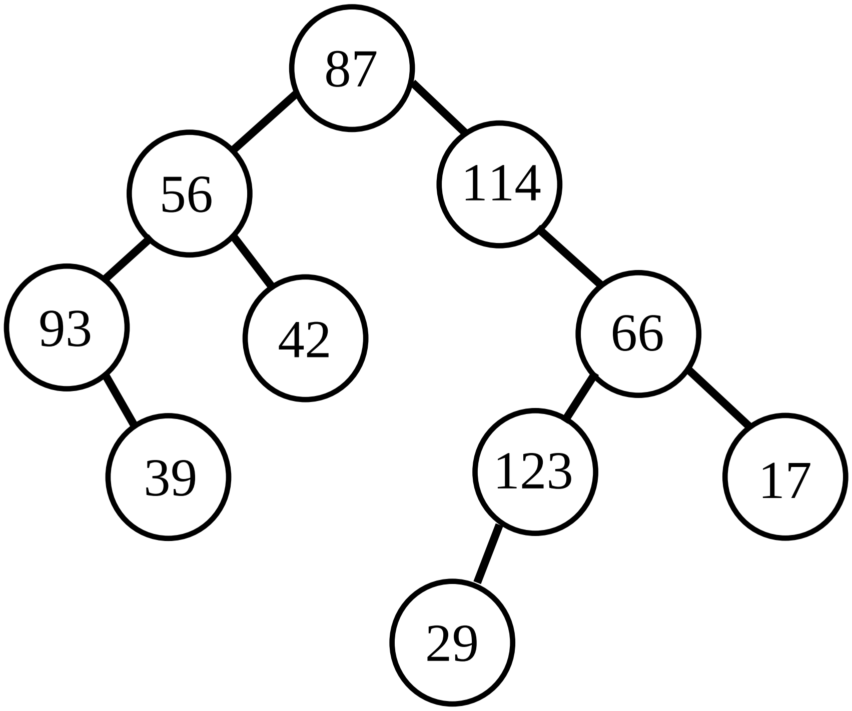
5. Prove that in any binary tree with n nodes there are $n + 1$ "null pointers".

6. Suppose that you have two traversals from the same binary tree. Draw the tree.
   pre-order: A D F G H K L P Q R W Z
   in-order: G F H K D L A W R Q P Z

7. Write a recursive member function of the BinaryTree class that counts the number of nodes in the tree.

8. Write a recursive member function of the BinaryTree class that counts the number of leaves in the tree.

9. Given the following binary tree containing integers, list the output from a *pre-order traversal*, an *in-order traversal*, a *post-order traversal*, and a *level-order traversal* of the tree.


87
56
114
93
42
66
39
123
17
29