Trees Worksheet

Name:____________________________________

1. What is the maximum number of nodes a binary tree of height h can have?

2. If a tree has n nodes, what is the minimum and maximum height the tree can have?

3. Should a binary search tree have the operations `insertAtEnd` or `insertAtIndex`? Explain why or why not.

4. What are the advantages of an *AVL Tree* or *Red-Black Tree* over an ordinary *Binary Search Tree*?

5. What are the disadvantages of an *AVL Tree* or *Red-Black Tree* over an ordinary *Binary Search Tree*?

6. What data structure is the Java *TreeSet* based on?

7. What can you tell me about the height of a *Red-Black Tree*?

8. What can you tell me about the height of a *AVL Tree*?

9. What is the worst-case running time of each of the following methods on the given data structures. Assume they are currently holding n elements. For the trees, assume the height is h unless you know otherwise (and in some cases you should know otherwise).

	BST	AVL Tree	Red-Black Tree	Linked List	Array
insert/add					
delete/remove					
search/contains					
maximum					
successor				N/A	N/A

10. Explain why in general a linked implementation for a binary tree is preferable to an array implementation.