BST Worksheet

In this worksheet we will practice the concepts of using a Binary Search Tree . For each of the following problems, draw the resulting Binary Search Tree.

1. Add the following numbers, in the order given to a binary search tree. 45, 67, 22, 100, 75, 13, 11, 64, 30
2. What is the height of the tree from #1?
3. What is the height of the subtree rooted at the node holding the value 22?
4. What is the depth of the node holding the value 22?
5. Add the following numbers, in the order given to a binary search tree. 3, 14, 15, 20, 25, 30, 33, 62, 200.
6. Is the tree from #3 balanced? Why not? What is the execution time required for searching for a value in this tree?
7. Add a new value, 145, to the tree from #1
8. Remove the value 67 from the tree from #1. What value did you replace it with and why?

Tree programming practice assignments

1. Write a recursive method that takes a reference to the root node of a tree T and returns a reference to the root node of the three that results from removing all leaves from T.
2. Write methods that take the root of a binary tree T as input, and calculate:
	1. The number of leaves in the tree,
	2. The number of nodes in T.

What is the running time of your routines?

1. Write a method that takes as input a binary search tree, T, and two keys, k1 and k2, which are ordered so that k1 <= k2, and prints all the elements in the tree such that: k1 <= Key(X) <= k2. Do not assume any information about the keys except that they can be ordered (consistently).