Practice Problems 4/25/2016

Part 1: Binary Search tree

1. Each node in a binary search tree contains data greater than all of the data in the node's _____ subtree.

2. Each node in a binary search tree contains data less than all of the data in the node's _____ subtree.

3. What is the efficiency (big O) of a binary search tree with height h and size n?

4. What is the efficiency (big O) of a full binary search tree with height h and size n?

5. What happens when you try to add a node to a tree that already exists in the tree?
a) It is added to the right subtree
b) it is added to the left subtree
c) It is not added to the tree
d) An IllegalArgumentException is thrown

6. Create the binary search tree when adding following nodes:
a) N A B M E R S P L C
b) F T N L E R S T A N
c) 4 3 9 10 6 1 2 5 7
d) 5 3 7 5 8 2 1 9 4 6

7. For the previous questions, what does the binary search tree look like after removing:
a) R
b) F
c) 9
d) 7

8. What are the possible cases for removing a Node from a tree?

9. When removing a node that has two children, which child becomes the root node?

Part 2: Clone

1. Does super.clone() make a shallow or a deep clone?

2. Is the following a shallow or deep clone of an array?

```
int[] listCopy = new int[list.length];
listCopy = list;
```

3. Is the following a shallow of deep clone of an array named *list*?

```
int[] listCopy = new int[list.length];
for (int i = 0; i < list.length; i++) {
     listCopy[i] = list[i];
}
```