Exercises
Binary Search Trees

1. The binary search tree provides us with a structure that allows us O(__) access to any node in the structure - an improvement over the sequential search of a(n) __ (list which is O(n)).
2. A binary tree is a structure in which each node is capable of having __ successor nodes, called __. The unique starting node is called the __.
3. A __ node is one that has no children. A node of a binary tree is itself the root of a smaller tree called a __. All nodes appearing in a subtree are called __ of the root node of the tree - conversely, the root node is called an __ of all nodes appearing below it. The root node of a tree is said to have level __. The maximum level in a tree determines its __ and the level contains at most __ nodes.
4. In a binary search tree, what is true of all nodes in the right subtree of some given node?
5. If you want to get rid of an existing tree, the statement `tree = NULL` is not recommended. Why?
6. Any node inserted in a binary search tree necessarily comes a __ node.
7. Minimizing the height of a Binary Search Tree will maximize
8. What three cases are considered in developing the Remove function?
9. Name the three BST traversals.
10. Given the character input (M G B H S P F C), draw the BST and state the tree traversals.
11. In destroying a tree which traversal should be used?
12. Which BST traversal lists the info fields of the tree nodes in sorted fashion?

Consider the tree below.

```
     G
    / \
   C   S
  / \   X
 /   |
M    P
```

13. How many subtrees are shown?
14. Insert nodes containing T and N. Redraw the tree.
15. After part 14, the trees height is __
16. Name the ancestors of P.
17. Interchange all subtrees with their corresponding right subtrees by redrawing the tree again.