Binary Search Trees Quiz (10pts)

1. (3pts) Draw the binary search tree that results from inserting the following values into an initially empty binary search tree in the following order: 54, 29, 19, 99, 42, 78, 44, 56, 97, 3, 13, 31, 47, 89

![Binary Search Tree Diagram]

-1 pt for each error, up to 3 errors.

2. (2pts) What are the outputs of a pre-order and post-order traversal of the final binary search tree drawn in question 1?

**Pre-order:** 54, 29, 19, 3, 13, 42, 31, 44, 47, 99, 78, 56, 97, 89
1 pt if completely correct OR 1 error, 0 otherwise.

**Post-order:** 13, 3, 19, 31, 47, 44, 42, 29, 56, 89, 97, 78, 99, 54
1 pt if completely correct OR 1 error, 0 otherwise.

3. (2pts) If a search was conducted for the value 48 in the final binary search tree from question #1, which nodes would get visited? (List them in the order they get visited.)

**54, 29, 42, 44, 47**
2 pts if completely correct, -1 pts if 1 error, otherwise 0 pts.
4. (3pts) Write a function which returns the smallest value stored in a *non-empty* binary search tree. The prototype is below:

```c
int minVal(struct treenode* root) {
    if (curr == NULL) // Grading – they don’t need
        return -1; // these lines, since it says
                      // non-empty binary search tree
    if (curr->left == NULL) // 1 point – completely correct
        return curr->data; // 1 point – completely correct
    return minVal(curr->left); // 1 point – completely correct
}
```