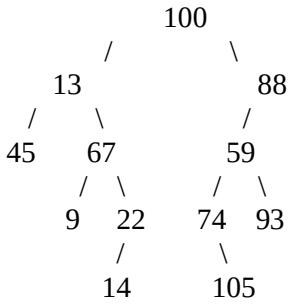
COP 3502 Quiz #4 Version A (Binary Search Trees, AVL Trees, Tries)

Last Name: _____, First Name: _____

Circle Your Recitation: F12:30 F1:30 F2:30 F3:30 F4:30

1) (6 pts) Provide the Preorder, Inorder and Postorder traversals of the following binary tree:

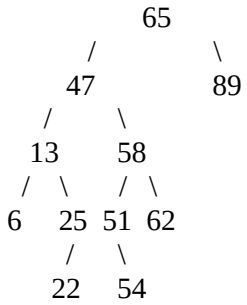


Pre-Order: ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___

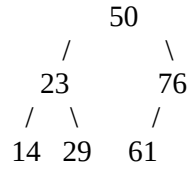
In-Order: ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___

Post-Order: ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___ , ___

2) (5 pts) Show the result of deleting 47 from the binary search tree shown below. (Note: There are two possible right answers.)



3) (9 pts) In a binary search tree, consider adding all the values at an even depth from the root, and subtracting all the values at an odd depth from the root. For example, for the tree shown below:

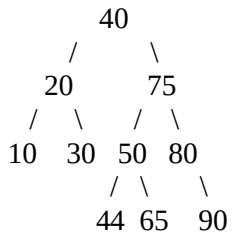


The desired sum would be 50 - 23 - 76 + 14 + 29 + 61 = 10.

Write a recursive function that calculates this adjusted sum for a binary search tree, given a pointer to its root. **(Hint: If I am a tree rooted at 50 above, then the contribution to my sum is the NEGATIVE of the corresponding sum for each of my subtrees.)** Please use the struct and function prototype given below:

```
typedef struct bintreenode {
    int data;
    struct bintreenode* left;
    struct bintreenode* right;
} bintreenode;

int getAdjustedSum(bintreenode* root) {









}
```

4) (5 pts) Show the result of inserting the value 60 into the AVL tree below. Put a box around your final answer.



5) (10 pts) Complete the function below so that it counts the number of words in the trie pointed to by root and returns this value. (Note: isWord is guaranteed to be either 0 or 1 for all nodes.) Please use the struct definition and function prototype shown below:

```
typedef struct trie {
    int isWord;
    struct trie* next[26];
} trie;

int numWords(trie* root) {









}
```

Scratch Page - Please clearly label any work on this page you would like graded.