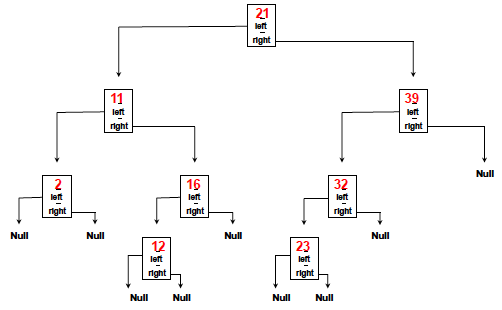
CS2050 –C Programming Quiz 9 Spring 2017  
DO NOT PUT YOUR ANSWERS ON THIS SHEET – RECORD THEM ON THE ANSWER SHEET ONLY.

1. a. Write out the INORDER traversal sequence for the tree diagram below.

b. Write out the PREORDER traversal sequence for the tree diagram below.



2. T / F The simple insertion algorithm we discussed in class and used in labs does not guarantee that the resulting BST is balanced.  
  
3. What is the following code designed to do with a BST?  
**void xxx(struct tree\_node \* p)  
 { if (p !=NULL) {  
 printf(“%d\n”, p->data);  
 xxx(p->left\_child);  
 xxx(p->right\_child);  
 }  
 }**

You’ve never seen this code before. Answer the questions below using the knowledge you’ve gained with nodes and trees. IGNORE COMPILE ERRORS!

x. void fred (treeT \*tptr, int key) {

b. treeT t;

c. int sign;

d. t = \*tptr;

e. if (t == NULL {

f. t = New(treeT);

g. t->key = key;

h. t->left = t->right = NULL;

m. \*tptr = t;

n. return;

o. }

p. if (key < prevkey)

r fred(&t->left, key);

s. else

t. fred(&t->right, key);

w. }

4. T / F Recursion is not used in this code.  
5. Where in this code (line number) is memory likely allocated?

6. (2 points) Based on the limited information you have, write out the struct definition for the data structure in use.

7. (2 points) What is this code designed to do?

CS2050 – C Programming Quiz 9 NAME \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Spring 2017 LAB SECTION \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

ANSWERS

1. a. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

b. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. T / F

3. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. T / F

5. \_\_\_\_\_\_\_\_\_\_\_\_\_

6. (2 points)

7. (2 points) \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_