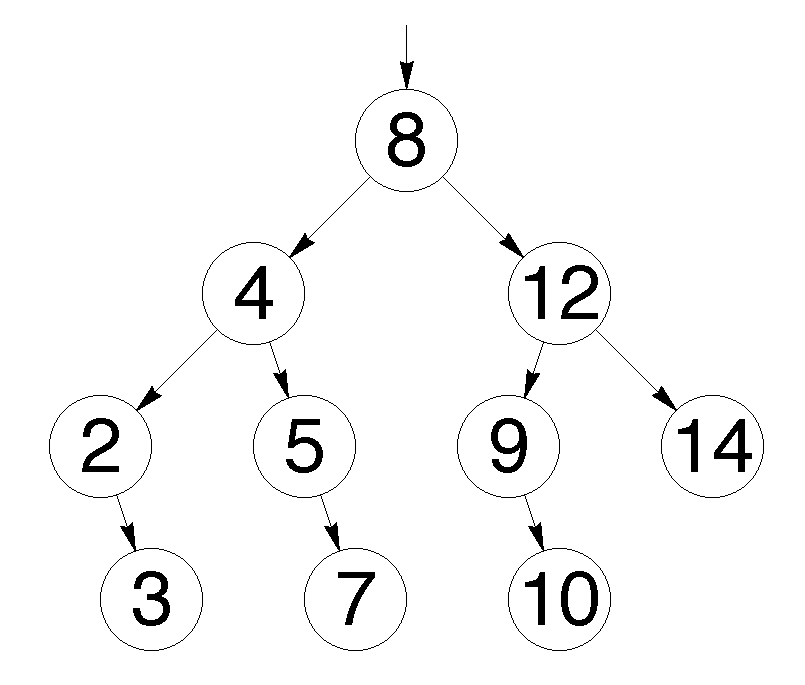
CS2050 –C Programming Quiz 8 Spring 2017  
DO NOT PUT YOUR ANSWERS ON THIS SHEET – RECORD THEM ON THE ANSWER SHEET ONLY.  


1. (2 points) Write out the INORDER traversal sequence for the tree diagram above.

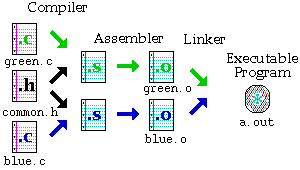
2. (2 points) Write out the PREORDER traversal sequence for the tree diagram above.

BIG O Notation:

3. (1 point) A perfectly balanced BST can be constructed directly from a sorted array in O (\_\_) time.

4. (1 point) The content-specific ordering of a BST allows it to satisfy search operations efficiently and can be represented with big O notation as O (\_\_ ).  
  
5. (2 points) What is the following code designed to do with a BST?  
 **struct treenode\* fred( int d , struct treenode \* p) {  
 if( p== null) {  
 p = (struct treenode\*)(malloc(sizeof(struct treenode)));  
 p->data = d;  
 p->left = NULL;  
 p->right = NULL;  
 }  
 else if(d < p->data)  
 p->left = fred( d, p->left);  
 else if(d > p->data)  
 p->right = fred( d, p->right );  
 return p;  
 }**

**Use the diagram below to answer the next several questions.**



6. (1 point) T / F The command to produce the an executable file named “finalProject” from the diagram above would be **cc green.c blue.c –o finalProject**

7. (1 point) T / F When you separate your C program into numerous files that allows you the flexibility to have functions with the same name.

CS2050 – C Programming Quiz 8 NAME \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

ANSWERS

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

2. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

4. \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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

6. T / F

7. T / F