**CS 249**

**Worksheet #5**

**4 November, 2014**

By

Brandon Horner

Part I

List the names in the tree below in the order they will be displayed using (a) an inorder traversal, (b) a preorder traversal, (c) a postorder traversal, and (d) a level order traversal.

Michael

/ \

John Joseph

/ \ / \

Jimmy Mary Carol Susan

(a) inorder traversal = Jimmy, John, Mary, **Michael**, Carol, Joseph, Susan

(b) preorder traversal = **Michael**, John, Jimmy, Mary, Joseph, Carol, Susan

(c) postorder traversal = Jimmy, Mary, John, Carol, Susan, Joseph, **Michael**

(d) level order traversal = **Michael**, John, Joseph, Jimmy, Mary, Carol, Susan

Part II

As we discussed in class there are multiple ways to traverse a tree. Using the Tree and Node classes given in Chapter #8 of your text write methods to display the nodes of a tree for each of the following traversals:

* inorder

public void inOrder(Node root){

if (node == null)

return;

inOrder(root.leftChild());

System.out.print(root.iData() + “, ”);

inOrder(root.rightChild());

}//end inOrder

* preorder

public void preOrder(Node root){

if (node == null)

return;

System.out.print(root.iData() + “, ”);

preOrder(root.leftChild());

preOrder(root.rightChild());

}//end preOrder

* postorder

public void postOrder(Node root){

if(root==null)

return;

postOrder(root.leftChild());

postOrder(root.rightChild());

System.out.print(root.iData() + “, ”);

}//end postOrder

* level order

public void levelOrder(int height) {

for (int i = 1; i <= height; i++) {

recLevelOrder(root, i);

}//end for

}//end levelOrder

public void recLevelOrder(Node root, int level) {

if (root == null)

return;

else if (level == 1)

System.out.print(root.iData + “ “);

else (level > 1) {

recLevelOrder(root.leftChild, level - 1);

recLevelOrder(root.rightChild, level - 1);

}//end else

}//end recLevelOrder

Note: Since the methods are to be implemented as part of the tree class, you might think that parameters aren’t needed since you have access to the root node. However, you should use recursion to implement these methods and thus, you will need to pass in the “local” root node as a parameter.