NAME: JAMES WASHINGTON

COURSE: DATA STRUCTURES

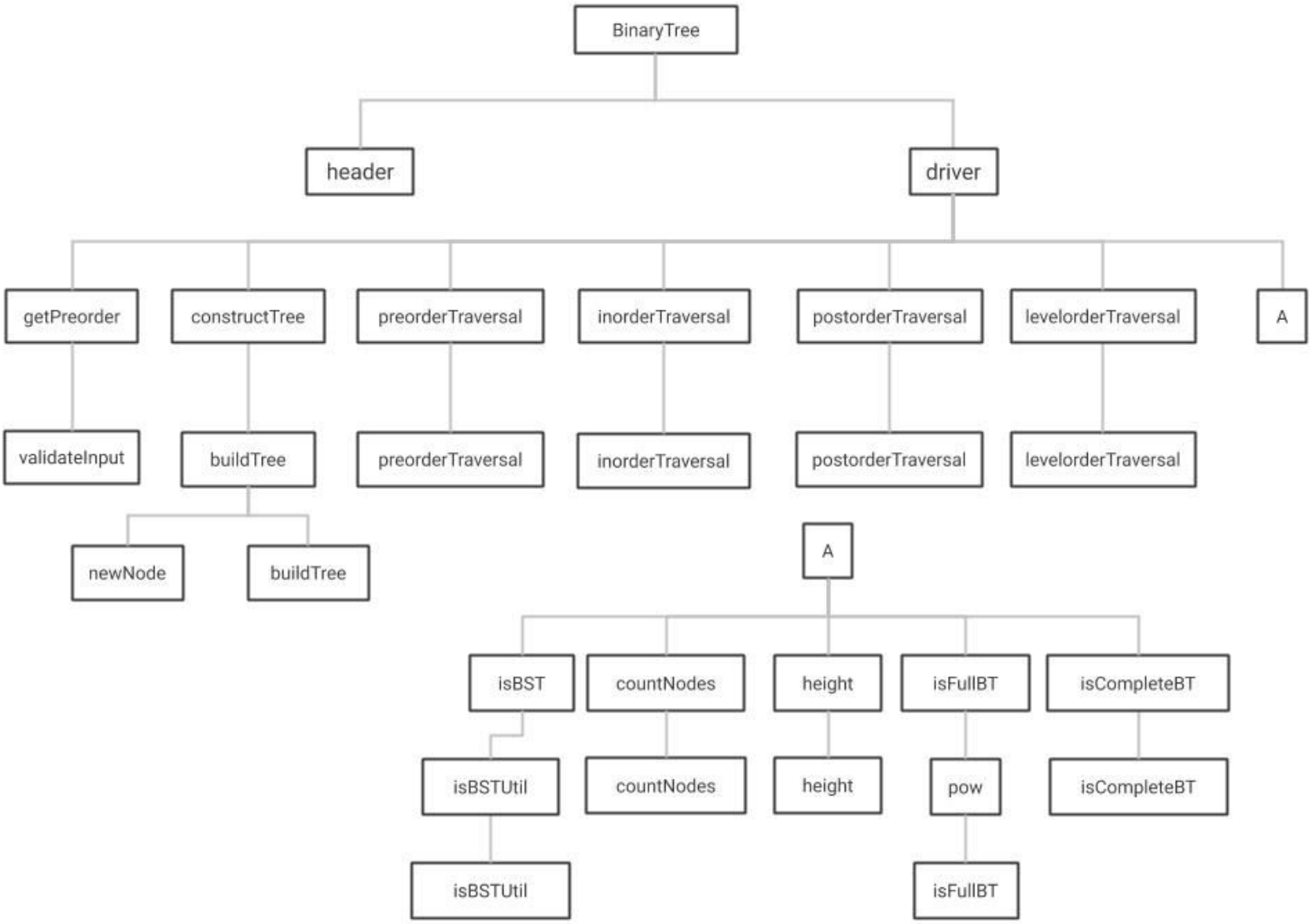
DUE DATE: 4/27/20

PROFESSOR: DR.NG

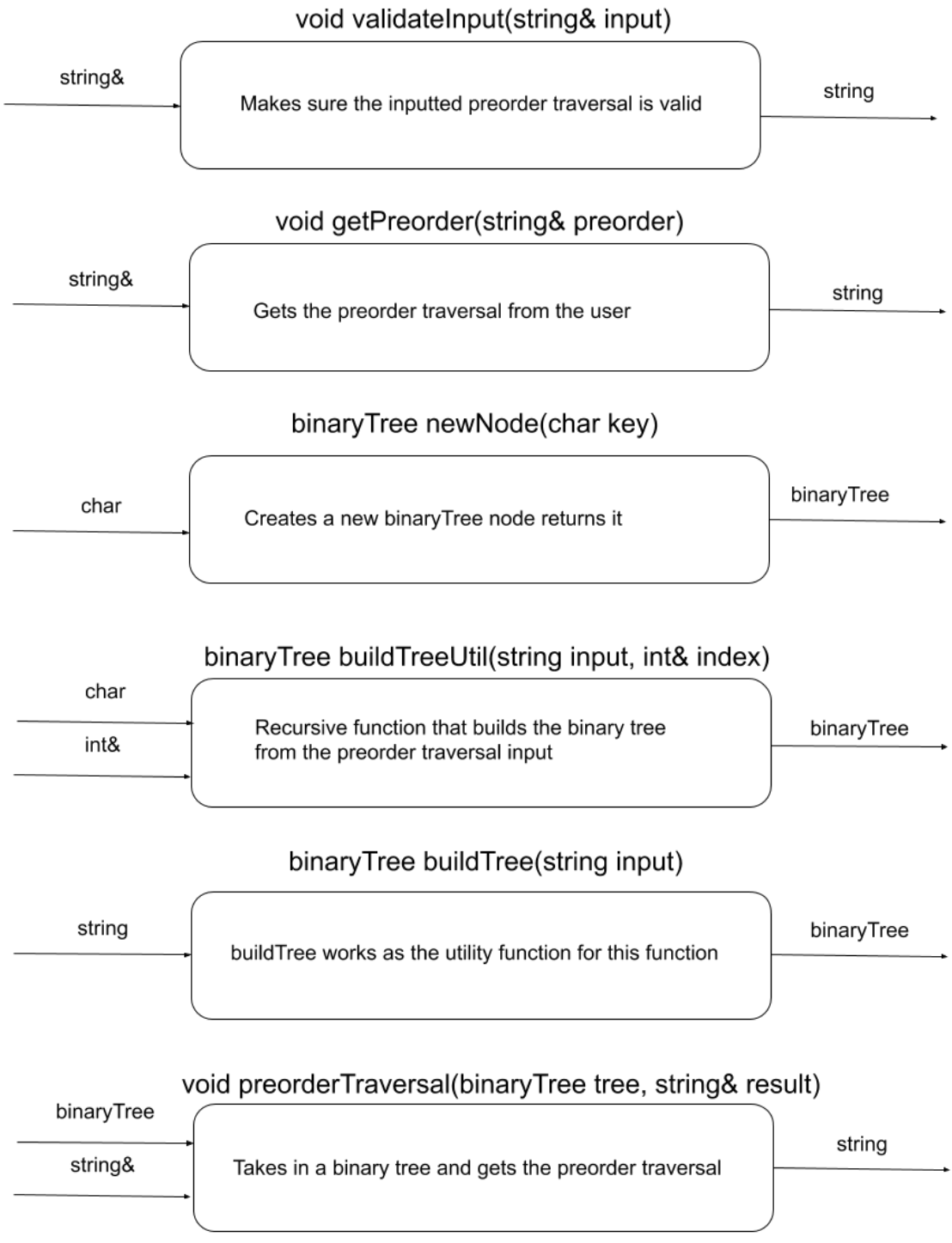
Program Design

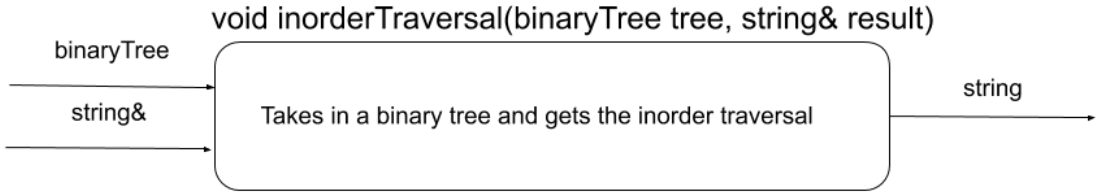
Declares variables to hold the results and the user input
Calls the getPreorder function so the user can input a preorder sequence
Calls the constructTree function to build the binary tree from the preorder sequence
Calls the preorderTraversal function to get the preorder of the tree
Calls the inorderTraversal function to get the inorder of the tree
Calls the postorderTraversal function to get the postorder of the tree
Calls the levelorderTraversal function to get the levelorder of the tree
Calls the isBST function to determine if the tree is a binary search tree
Calls the countNodes function to get the number of nodes in the tree
Calls the height function to get the height of the tree
Calls the isFullBT function to determine the tree is full
Calls the isCompleteBT to determine if the tree is complete
Output all the traversals of the tree
If (tree is a binary search tree)
    Output result
Output the number of nodes in the tree
Output the height of the tree
If (tree is a full binary tree)
    Output results
If (tree is a complete binary tree)
    Output results

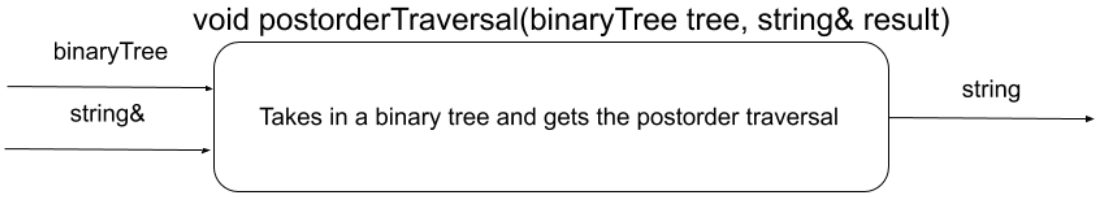
Hierarchical Diagram

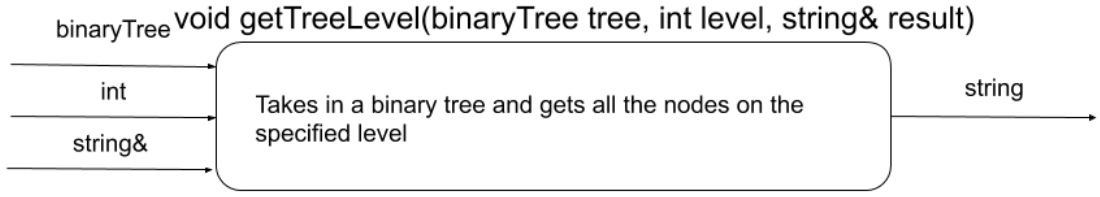


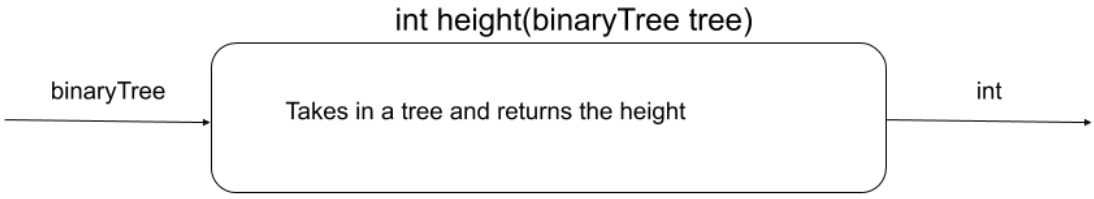
Procedure Specification

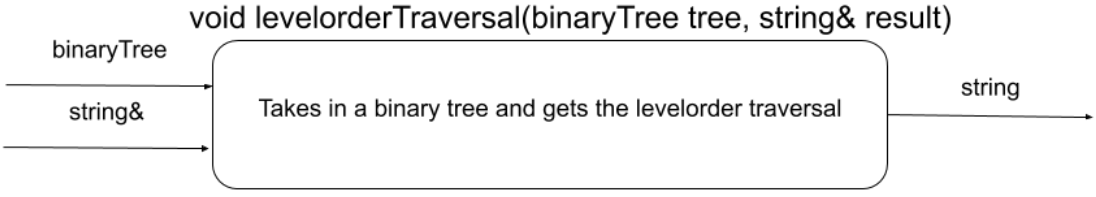


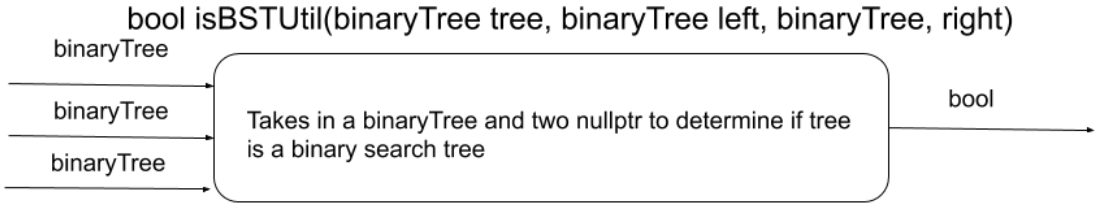

void inorderTraversal(binaryTree tree, string& result)
binaryTree
string&
Takes in a binary tree and gets the inorder traversal
string


void postorderTraversal(binaryTree tree, string& result)
binaryTree
string&
Takes in a binary tree and gets the postorder traversal
string

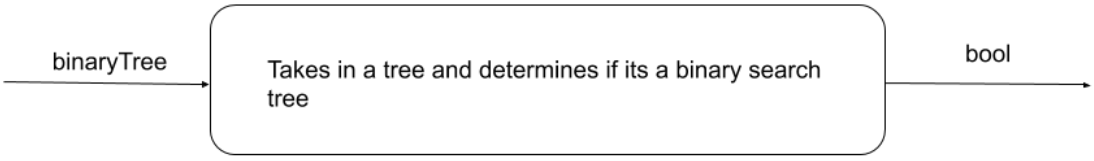

void getTreeLevel(binaryTree tree, int level, string& result)
binaryTree
int
string&
Takes in a binary tree and gets all the nodes on the specified level
string


int height(binaryTree tree)
binaryTree
Takes in a tree and returns the height
int

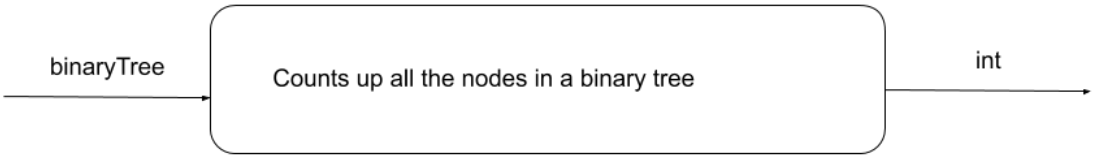

void levelorderTraversal(binaryTree tree, string& result)
binaryTree
string&
Takes in a binary tree and gets the levelorder traversal
string


bool isBSTUtil(binaryTree tree, binaryTree left, binaryTree, right)
binaryTree
binaryTree
binaryTree
Takes in a binaryTree and two nullptr to determine if tree is a binary search tree
bool

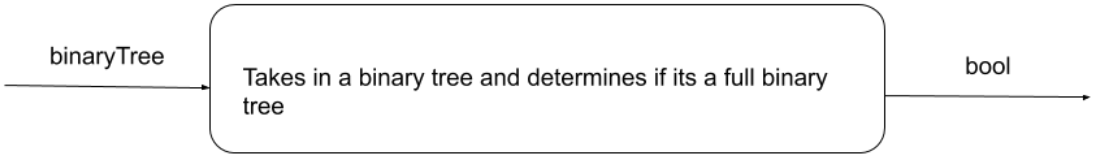
bool isBST(binaryTree tree)



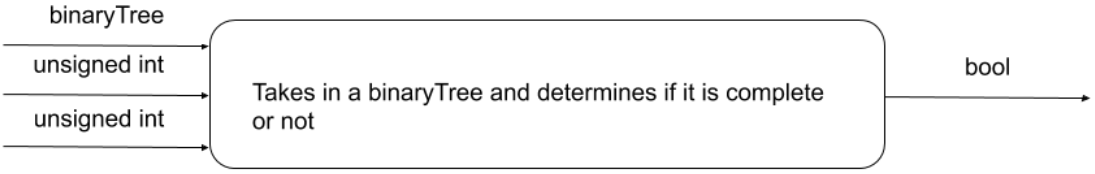
int countNodes(binaryTree tree)



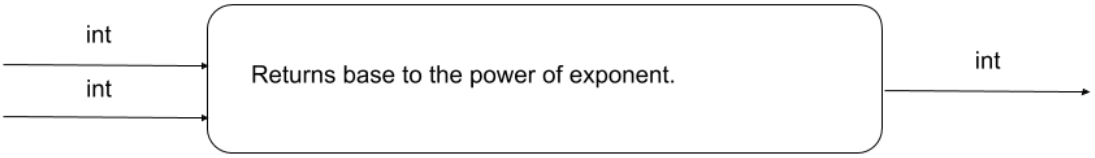
bool isFullBT(binaryTree tree)



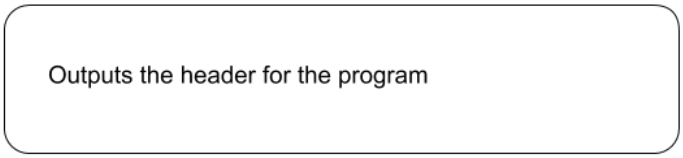
bool isCompleteBT(binaryTree tree, unsigned int index, unsigned int numOfNodes)



int pow(int base, int exponent)



void header( )

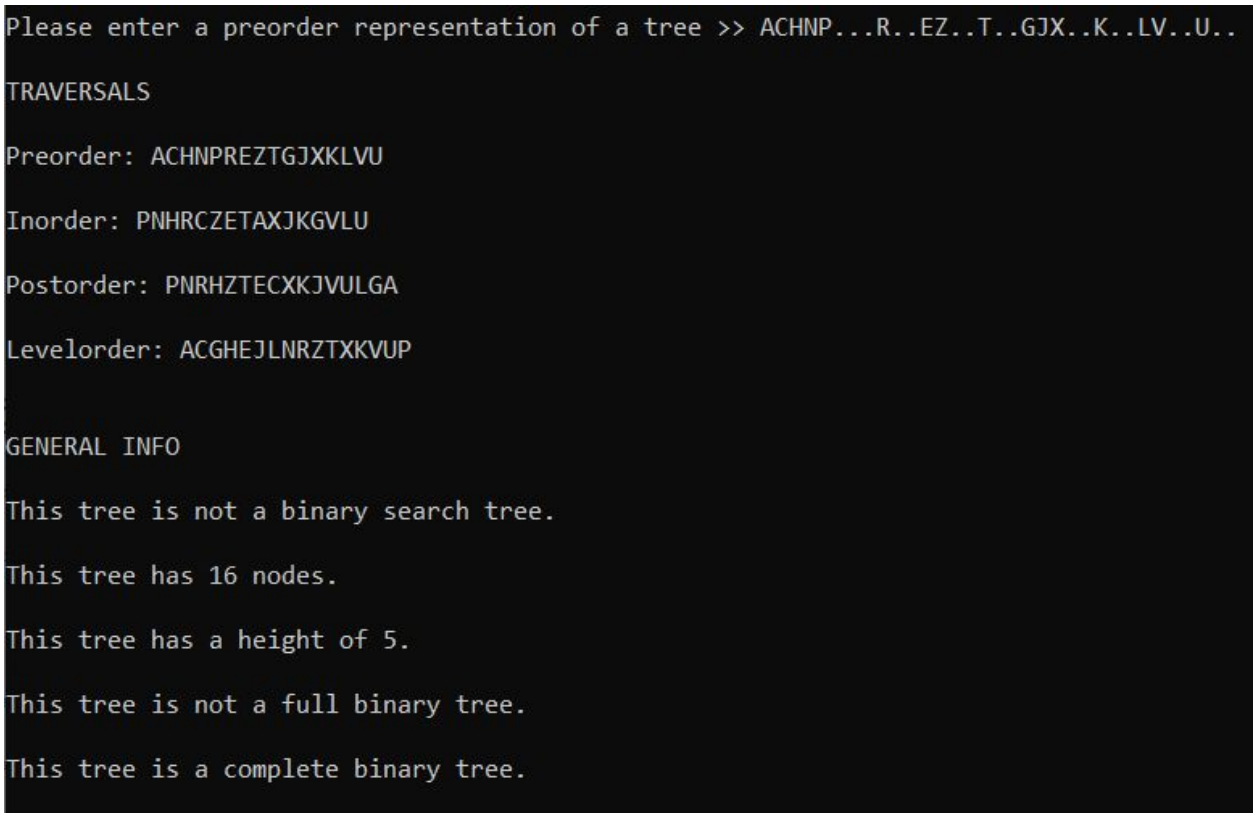


void driver( )

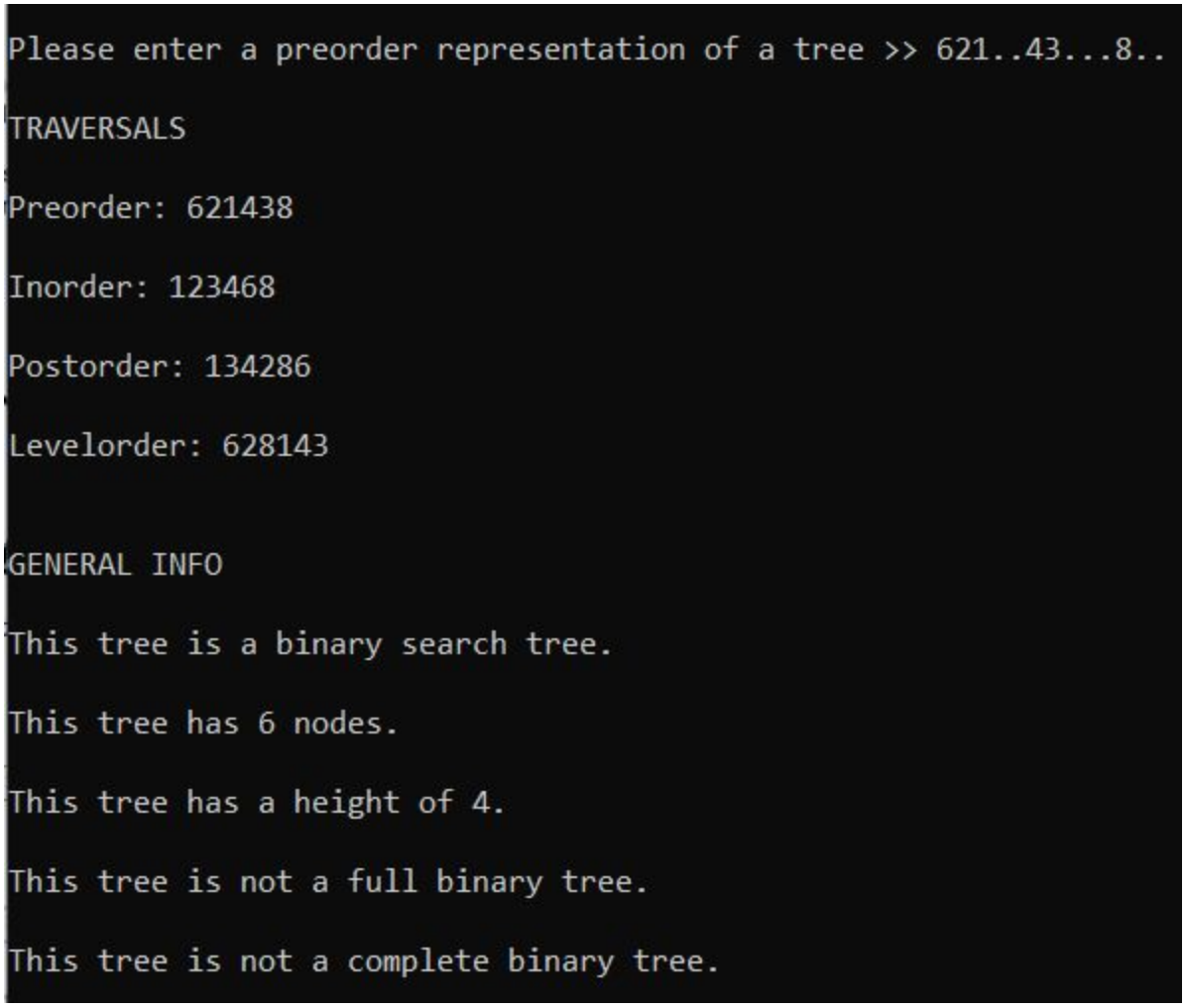
Program Testing

```
Please enter a preorder representation of a tree >> *5 9 .. 71 . . .4 .$ . .

TRAVERSALS

Preorder: *59714$

Inorder: 9517*4$

Postorder: 9175$4*

Levelorder: *5497$1


GENERAL INFO

This tree is not a binary search tree.

This tree has 7 nodes.

This tree has a height of 4.

This tree is not a full binary tree.

This tree is not a complete binary tree.
```

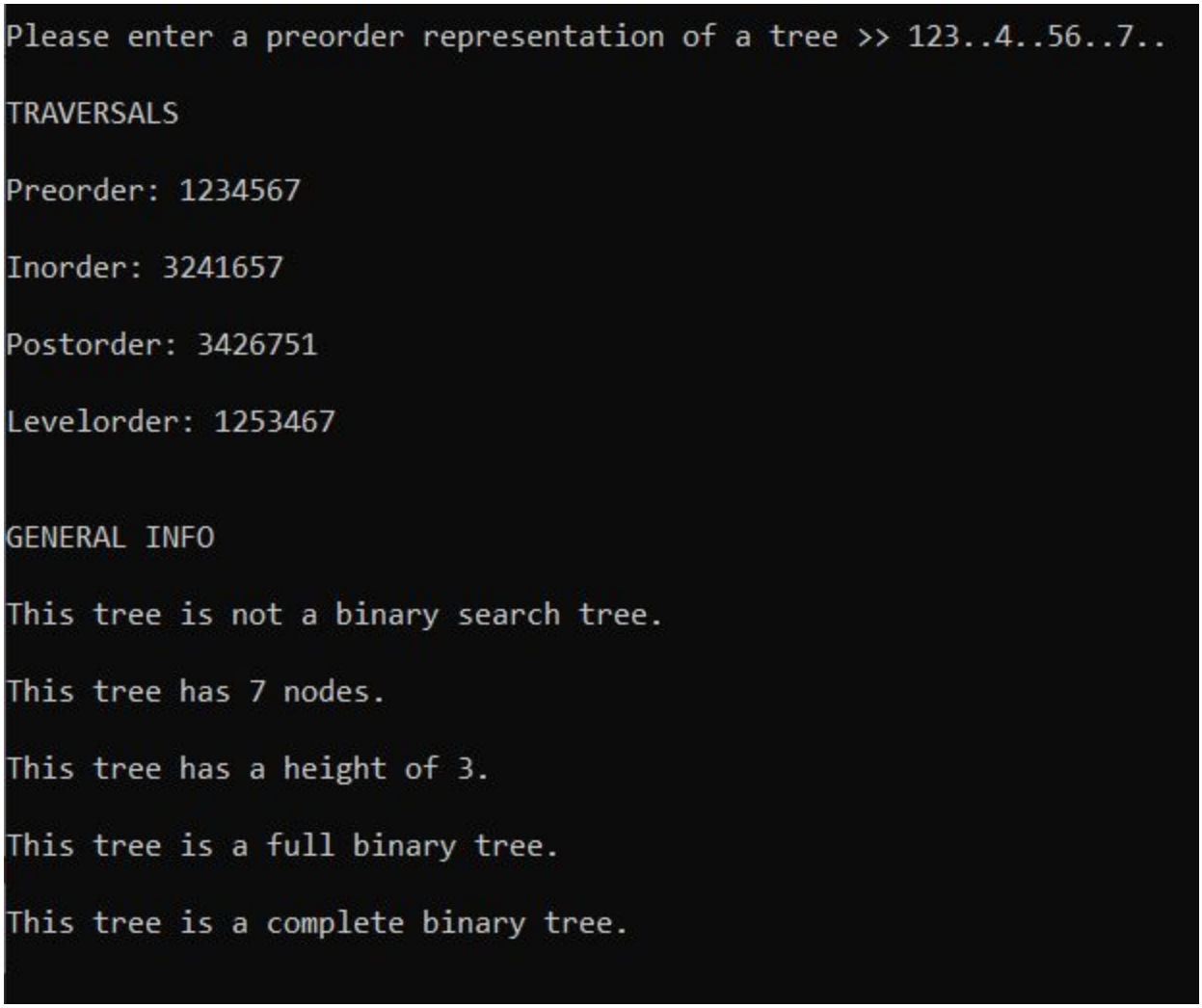
Tested to make sure special characters are allowed, and all the info is correct. Spaces are optional, I thought it would be better for the user that way even if it wasn't in the specifications.



Tested with a bigger complete tree, and made sure all the information is correct.



Tested with a binary search tree and made sure all the information was correct



Tested with a full binary tree to make sure it worked properly