

CS 211 Lab 10

Part 1(copied from vscode plz give credit)

```
bool check = true;
void preorderTraversal(Node* root, bool& check){
    if(root == null){
        return;
    }

    Node* left = root->left;
    Node* right = root->right;

    if(left != null){
        if(left->data > root->data ){
            check = false;
        }
    }
    if(right != null){
        if(right->data < root->data){
            check = false;
        }
    }

    checkBST(root->left, check);
    checkBST(root->right, check);
}
```

Part2

Need to be able to traverse from start to end

Use breadth first search

Use queue and set to keep track of visited nodes

If start to end can be traversed then return true

Otherwise false

```
bool canReach(neighborList, Node* start, Node* end){
```

```
if(start == end){
    return true;
}

Queue queue;
queue.enqueue(start);

set visited;
visited.add(start);

while (!queue.isEmpty())
{
    Node* current = queue.dequeue();

    for(auto i : neighborList[current]){
        if(current == end){
            return true;
        }
        if(visited.count(current) == false){
            visited.add(current);
        }
    }
}
return false;
}
```