#pragma once

#include <stdio.h>

#include <stdlib.h>

#include <limits.h>

class BSTConfirmation

{

public:

BSTConfirmation();

~BSTConfirmation();

struct node {

int data;

struct node\* left;

struct node\* right;

};

int bstCheck(struct node\* node, int min, int max) {

if (node == NULL)

return 1;

//Cannot exceed min or max values

if (node->data < min || node->data > max)

return 0;

//Recursively check all subtree's

return

bstCheck(node->left, min, node->data - 1) &&

bstCheck(node->right, node->data + 1, max);

};

int isBST(struct node\* node) {

//Start with all possible int values for min and max

return bstCheck(node, INT\_MIN, INT\_MAX);

};

struct node\* createNewNode(int data)

{

//Allocate space needed for new node

struct node\* node = (struct node\*)

malloc(sizeof(struct node));

node->data = data;

node->left = NULL;

node->right = NULL;

return(node);

};

int main()

{

struct node\* root = createNewNode(4);

//Manually assign left and right pointers

root->left = createNewNode(2);

root->left->left = createNewNode(1);

root->left->right = createNewNode(3);

root->right = createNewNode(5);

root->right->left = createNewNode(6);

root->right->right = createNewNode(7);

if (isBST(root))

printf("Is BST");

else

printf("Not a BST");

return 0;

};

};