#include <bits/stdc++.h>

using namespace std;

class Node {

public:

int data;

Node \*left;

Node \*right;

Node(int d) {

data = d;

left = NULL;

right = NULL;

}

};

class Solution {

public:

void preOrder(Node \*root) {

if( root == NULL )

return;

std::cout << root->data << " ";

preOrder(root->left);

preOrder(root->right);

}

/\*

Node is defined as

class Node {

public:

int data;

Node \*left;

Node \*right;

Node(int d) {

data = d;

left = NULL;

right = NULL;

}

};

\*/

Node \* insert(Node \* root, int value) {

if(root==NULL) {

Node\* newNode;

newNode = (Node\*)malloc(sizeof(Node));

newNode->left = NULL;

newNode->right = NULL;

newNode->data = value;

return newNode;

}

if(value <= root->data)

root->left = insert(root->left, value);

else

root->right = insert(root->right, value);

return root;

}

};

int main() {

Solution myTree;

Node\* root = NULL;

root = myTree.insert(root, 10);

root = myTree.insert(root, 20);

root = myTree.insert(root, 1);

root = myTree.insert(root, 4);

root = myTree.insert(root, 67);

root = myTree.insert(root, 50);

myTree.preOrder(root);

return 0;

}