#include <iostream>

#include<climits>

using namespace std;

class node{

public:

int data;

node \*left;

node \*right;

node(int d){

data=d;

left=NULL;

right=NULL;

}

};

node\* insertInBST(node \*root,int data){

if(root==NULL){

return new node(data);

}

if(data<=root->data){

root->left=insertInBST(root->left,data);

}

else{

root->right=insertInBST(root->right,data);

}

return root;

}

node\* build(){

int d;

cin>>d;

node \*root=NULL;

while(d!=-1){

root=insertInBST(root,d);

cin>>d;

}

return root;

}

//inorder of bst is always sorted

void inorder(node \*root){

if(root==NULL)

return;

inorder(root->left);

cout<<root->data<<", ";

inorder(root->right);

}

bool isBST(node \*root,int min=INT\_MIN,int max=INT\_MAX){

if(root==NULL){

return true;

}

if(root->data>=min && root->data<=max && isBST(root->left,min,root->data) && isBST(root->right,root->data,max)){

return true;

}

return false;

}

int main()

{

node \*root=build();

inorder(root);

cout<<endl;

if(isBST(root)){

cout<<"Yes";

}

else{

cout<<"No";

}

return 0;

}

Input-

5 3 7 1 6 8 -1

Output-

1,3,5,6,7,8,

Yes