#include<iostream>

#include<stack>

using namespace std;

//DEFINE A DATATYPE FOR NODE

struct node{

int price;

string brand;

string fuel;

string car\_type;

node \*lchild;

node \*rchild;

};

//CREATE A NODE

node \*getnode(){

node \*temp;

temp=new node;

cout<<"ENTER THE PRICE:";

cin>>temp->price;

cout<<"ENTER THE BRAND:";

cin>>temp->brand;

cout<<"ENTER THE FUEL TYPE:";

cin>>temp->fuel;

cout<<"ENTER THE CAR TYPE:";

cin>>temp->car\_type;

temp->lchild=NULL;

temp->rchild=NULL;

return temp;

}

node \*create\_tree(){

node \*temp,\*ptr,\*root=NULL;

char ch,dir;

do{

temp=getnode();

if(root==NULL)

root=temp;

else{

ptr=root;

while(ptr!=NULL){

cout<<"ENTER THE DIRECTION(L/R);";

cin>>dir;

if(dir=='l'){

if(ptr->lchild==NULL){

ptr->lchild=temp;

break;

}

else{

ptr=ptr->lchild;

}

}

else{

if(ptr->rchild==NULL){

ptr->rchild=temp;

break;

}

else{

ptr=ptr->rchild;

}

}

}

}

cout<<"WANT TO CONTINUE(Y/N):";

cin>>ch;

}while(ch=='y' || ch=='Y');

return root;

}

void preorder(node \*root){

stack<node\*> s;

node \*temp;

temp=root;

do{

while(temp!=NULL)

{

cout<<temp->price<<"->";

cout<<temp->brand<<"->";

cout<<temp->fuel<<"->";

cout<<temp->car\_type<<endl;

if(temp->rchild!=NULL)

s.push(temp->rchild);

temp=temp->lchild;

}

if(!s.empty()){

temp=s.top();

s.pop();

}

}while(!s.empty() || temp!=NULL);

cout<<endl;

}

void inorder(node \*root){

stack<node\*> s;

node \*curr=root;

while(curr!=NULL || !s.empty())

{

while(curr!=NULL)

{

s.push(curr);

curr=curr->lchild;

}

curr=s.top();

s.pop();

cout<<curr->price<<"->";

cout<<curr->brand<<"->";

cout<<curr->fuel<<"->";

cout<<curr->car\_type<<endl;

curr=curr->rchild;

}

cout<<endl;

}

//POSTORDER TRAVERSAL PROGRAM

void postorder(node \*root){

if(root==NULL)

return ;

stack<node\*> s1,s2;

node \*curr=root;

s1.push(curr);

while (!s1.empty())

{

curr=s1.top();

s1.pop();

s2.push(curr);

if(curr->lchild!=NULL)

s1.push(curr->lchild);

if(curr->rchild!=NULL)

s1.push(curr->rchild);

}

while(!s2.empty()){

curr=s2.top();

cout<<curr->price<<"->";

cout<<curr->brand<<"->";

cout<<curr->fuel<<"->";

cout<<curr->car\_type<<endl;

s2.pop();

}

cout<<endl;

}

int main(){

node \*t;

t=create\_tree();

cout<<"PREORDER TRAVERSAL OF TREE:"<<endl;

preorder(t);

cout<<"INORDER TRAVERSAL OF TREE:"<<endl;

inorder(t);

cout<<"POSTORDER TRAVERSAL OF TREE:"<<endl;

postorder(t);

return 0;

}