**Binary tree using Linked List**

#include<iostream.h>

#include<conio.h>

struct Node

{

int data;

struct Node \*left, \*right;

Node(int data)

{

this->data=data;

left=right=NULL;

}

};

void preordertraversal(struct Node\* node)

{

if(node==NULL)

return;

cout<<node->data<<"->";

preordertraversal(node->left);

preordertraversal(node->right);

}

void postordertraversal(struct Node\* node)

{

if(node==NULL)

return;

postordertraversal(node->left);

postordertraversal(node->right);

cout<<node->data<<"->";

}

void inordertraversal(struct Node\* node)

{

if(node==NULL)

return;

inordertraversal(node->left);

cout<<node->data<<"->";

inordertraversal(node->right);

}

void main()

{

clrscr();

struct Node\* root=new Node(1);

root->left=new Node(12);

root->right=new Node(9);

root->left->left=new Node(5);

root->left->right=new Node(6);

cout<<"BINARY TREE USING LINKED LIST"<<endl;

cout<<"\nPreorder Traversal:"<<endl;

preordertraversal(root);

cout<<"NULL";

cout<<"\n";

cout<<"\nInorder Traversal:"<<endl;

inordertraversal(root);

cout<<"NULL";

cout<<"\n";

cout<<"\nPostorder Traversal:"<<endl;

postordertraversal(root);

cout<<"NULL";

cout<<"\n";

getch();

}

**Output:**

