```
#include<stdio.h>
#include<stdlib.h>
#include<iostream>
#include<queue>

using namespace std;

struct BinaryNode
{
    struct BinaryNode *left;
    struct BinaryNode *right;
    int data;
};

struct BinaryNode * createBinaryNode(int value)
{
    struct BinaryNode *B=(struct BinaryNode *)malloc(sizeof(struct BinaryNode));
    B->left=NULL;
    B->right=NULL;
    B->data=value;
    return B;
}
queue<BinaryNode *> q;
int leftViewOfTree(struct BinaryNode *root)
{
    int flag=0;
    if(!root)
        return 0;
    q.push(root);
    q.push(NULL);
    printf("%d-->",root->data);
    while(!q.empty())
    {
        struct BinaryNode *temp=q.front();
        q.pop();
        if(temp==NULL)
        {
            flag=1;
            if(!q.empty())
                q.push(NULL);

        }
        else
        {
            if(flag==1)
            {
              printf("%d-->",temp->data);
              flag=0;
            }

            if(temp->left)
                q.push(temp->left);
            if(temp->right)
                q.push(temp->right);
        }
    }
}

int main()
{
    struct BinaryNode *root=createBinaryNode(10);
    root->right=createBinaryNode(20);
    root->left=createBinaryNode(30);
    root->left->left=createBinaryNode(40);
    root->left->right=createBinaryNode(50);
    root->right->right=createBinaryNode(60);
```

```
67      root->right->left=createBinaryNode(70);
68
69      printf("left view of tree is \t");
70      leftViewOfTree(root);
71  }
72
```