Name : Dev Adnani
SID : 202212012
Subject : DSA
Topic : Binary Trees
Lab : 05

```
#include <iostream>
#include <queue>
using namespace std;

class Node
{
public:
	int data;
	Node *left;
	Node *right;

	Node(int data)
	{
	this->data = data;
	this->left = NULL;

	this->right = NULL;
	}
};
```

```
Node *bt(Node *head)
{

	int data;
	cin >> data;

	if (data == -1)
	{
	return NULL;
	}

	head = new Node(data);
	cout << "Enter the data in left of " << data << endl;
	head->left = bt(head->left);
	cout << "Enter the data in right of " << data << endl;
	head->right = bt(head->right);

	return head;
}
```

```
void preorder(struct Node *rt)
{
	if (rt == NULL)
	{
	return;
	}

	cout << rt->data << " ";
	preorder(rt->left);
	preorder(rt->right);
}
```

```
void inorder(struct Node *rt)
{
	if (rt == NULL)
	{
	return;
	}
	inorder(rt->left);

	cout << rt->data << " ";
	inorder(rt->right);
}

void postorder(struct Node *rt)
{
	if (rt == NULL)
	{
	return;
	}
	postorder(rt->left);
	postorder(rt->right);
	cout << rt->data << " ";
}
```

```
int sumOfLeafNodes(Node *root)
{
	if (root == NULL)
	{
	return -1;
	}

	queue<Node *> q;
	q.push(root);
	q.push(NULL);
	int sum = 0;

	while (!q.empty())
	{
	Node *node = q.front();
	q.pop();

	if (node == NULL)
	{
		cout << sum << endl;
		if (!q.empty())
		{
		q.push(NULL);
		sum = 0;
		}
	}

	else
```

```
        {
            if (node->left == NULL && node->right == NULL)
            {
            sum = sum + node->data;
            }

            if (node->left)
            {
            q.push(node->left);
            }

            if (node->right)
            {
            q.push(node->right);
            }
        }
        }
}
```

```
int main()
{

	Node *rt = NULL;
	cout << "Enter -1 to add NULL " << endl;
	cout << "Enter the data: " << endl;
	rt = bt(rt);

	cout << endl
	<< "Preorder : " << endl;
	preorder(rt);
	cout << endl
	<< "Inorder : " << endl;
	inorder(rt);
	cout << endl
	<< "Postorder : " << endl;
	postorder(rt);


	cout<<endl <<" Sum Of Leaf Nodes :" <<endl
	<<sumOfLeafNodes(rt)<<endl;

	return 0;
}
```

O/P :


Preorder :
1 2 4 8 9 5 10 11 3 6 13 7 14
Inorder :
8 4 9 2 10 5 11 1 6 13 3 14 7
Postorder :
8 9 4 10 11 5 2 13 6 14 7 3 1 0
0
0
65
Sum Of Leaf Nodes :
6422144