Name : Swami Amit Bandu

Class :- SE COMP

Roll No. :- 2221018

Ass 3

Problem:- Convert given binary tree into threaded binary tree. Analyze time and space complexity of the algorithm.

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

class node
{
    public:
       int data,lb,rb;
        node *left,*right;
};

class thread
{
    public:
       void create(node *head);
       void inorder(node *head);
};

void thread::create(node *head)
{
     node *root,*ne,*temp;
     root =new node;
     int n;char ans;
     cout<<"Enter root data :";
     cin>>root->data;
     root->left=head;
     root->right=head;
     head->left=root;
     head->lb=1;
    do{
    ne=new node;
   cout<<"Enter new data :";
   cin>>ne->data;
ne->left=ne->right=NULL;
```

```
ne->lb=ne->rb=0;
temp=root;
while(1)
{
   cout<<" Enter left or right(l/r) of "<<temp->data<<" :";
   cin>>ans;
   if(ans=='l')
  {
  if(temp->lb==0)
  {
   ne->left=temp->left;
   ne->right=temp;
   temp->left=ne;
   temp->lb=1;
   break;
 }
else
 {
    temp=temp->left;
 }

}
else
{
     if(temp->rb==0)
    {
       ne->right=temp->right;
       ne->left=temp;
       temp->right=ne;
      temp->rb=1;
      break;
  }
  else
 {
     temp=temp->right;
 }


}

 }
```

```
    cout<<"DO you want to continue(1/0) :";
    cin>>n;
     }while(n==1);
}

void thread::inorder(node *head)
{
node* temp;
temp=head->left;
while(temp!=head)
    {
    while(temp->lb==1)
    {
      temp=temp->left;
      }
    cout<<temp->data;
   while(temp->rb==0)
   {
  temp=temp->right;
   if(temp==head)
   break;
        cout<<temp->data;
  }
    temp=temp->right;
 }

}

int main()
{
thread ob;
node *head;
int ch;
while(1)
{
cout<<"\n1. Create \n";
cout<<"2. Inorder \n";
cout<<"Enter your choice :";
cin>>ch;
switch(ch)
{
```

```
case 1:head=new node;
           head->left=head->right=head;
          head->lb=head->rb=0;
          ob.create(head);
         break;

case 2:ob.inorder(head);
          break;
       }

   }

}
```