/\*

***Name* :-** Abrar Mahedavi

***Roll No.* :-** 21035

***Title* :-** Write a C++ program to implement traversal on threaded binary tree using object oriented programming features. Design necessary class.

**50**

**- -**

**- -**

**30 60**

**- - - -**

**- - - -**

**15 40 55 70**

for above tree **inorder** traversal:- 1**5 30 40 50 55 60 70**

for above tree **pre order** traversal:- **50 30 15 40 60 55 70**

**Note:-** see the below implementation with comments consideing above example

*/\**

***Program :***

#include<iostream>

using namespace std;

class TBT

{

private:

int data;

int lbit,rbit;

TBT\*lchild,\*rchild;

public:

TBT\*create\_TBT(TBT\*,TBT\*);

void inorder\_TBT(TBT\*,TBT\*);

void preorder\_TBT(TBT\*,TBT\*);

};

void TBT::preorder\_TBT(TBT\*root,TBT\*header)

{

TBT\*trav;

int flag=0;

trav=root;

while(trav!=header)

{

while(trav->lbit==0&&flag==0) //step 1) if normal child

{

cout<<" "<<trav->data; // print 50 and then 30

trav=trav->lchild; // go to left child

}

if(flag==0) // step 2) if normal child but lbit=1 i.e last child

cout<<" "<<trav->data; // print 15 and then 40, 55 , 70

if(trav->rbit==0&&flag==0)

{

trav=trav->rchild;

}

else

{

if(trav->rbit==0&&flag==1)// step no 4 go to 40

{

trav=trav->rchild;

flag=0; // mark 40 as not visited

}

else

{

if(trav->rbit==1&&flag==0)//step 3) if rt child is thread climb up

{

trav=trav->rchild; //from 15 go to 30 again

flag=1; // make 30 as visited or printed

}

else

{

if(trav->rbit==1&&flag==1)

{

trav=trav->rchild;

}

}

}

}

}

}

void TBT::inorder\_TBT(TBT\*root,TBT\*header)

{

int flag=0;

TBT\*trav;

trav=root;

while(trav!=header)

{

while(trav->lbit!=1&&flag==0)

{

trav=trav->lchild; // go to left

}

cout<<" "<<trav->data; // ptint leftmost data

if(trav->rbit==0)

{

trav=trav->rchild;

flag=0;

}

else

{

trav=trav->rchild;

flag=1;

}

}

}

TBT\*TBT::create\_TBT(TBT\*root,TBT\*header)

{

TBT\*trav,\*temp,\*p;

int attached\_flag=0;

char ans;

while(1)

{

trav=root;

if(root==NULL)

{

temp=new TBT();

cout<<"\nEnter the data ";

cin>>temp->data;

temp->lbit=1;

temp->rbit=1;

temp->lchild=header;

temp->rchild=header;

root=temp;

}

else

{

temp=new TBT();

cout<<"\nEnter the data ";

cin>>temp->data;

temp->lbit=1;

temp->rbit=1;

attached\_flag=0;

trav=root;

while(attached\_flag==0)

{

if(trav->data<temp->data&&trav->rbit==0)

{

trav=trav->rchild;

}

else

if(trav->data<temp->data&&trav->rbit==1)

{

trav->rbit=0;

p=trav->rchild;

trav->rchild=temp;

temp->rchild=p;

temp->lchild=trav;

attached\_flag=1;

}

if(trav->data>temp->data&&trav->lbit==0)

{

trav=trav->lchild;

}

else

if(trav->data>temp->data&&trav->lbit==1)

{

trav->lbit=0;

p=trav->lchild;

trav->lchild=temp;

temp->lchild=p;

temp->rchild=trav;

attached\_flag=1;

}

}

attached\_flag=0;

}

cout<<"\nDo you want to attach more nodes [y/n] ";

cin>>ans;

if(ans=='n'||ans=='N')

break;

}

return root;

}

int main()

{

int choice,c;

TBT\*root=NULL,obj,\*header=NULL;

c=1;

while(c=1)

{

cout<<"\n1.Create TBT";

cout<<"\n2.Inorder TBT";

cout<<"\n3.Preorder TBT";

cout<<"\n4.Exit";

cout<<"\nEnter your choice :";

cin>>choice;

switch(choice)

{

case 1: root=obj.create\_TBT(root,header);

break;

case 2: obj.inorder\_TBT(root,header);

break;

case 3: obj.preorder\_TBT(root,header);

break;

case 4:c=0;

}

}

return 0;

}

***Output:***

compeng-sl2-08@compeng-sl2-08:~/Abrar$ g++ TBT.cpp

compeng-sl2-08@compeng-sl2-08:~/Abrar$ ./a.out

1.Create TBT

2.Inorder TBT

3.Preorder TBT

4.Exit

Enter your choice :1

Enter the data 50

Do you want to attach more nodes [y/n] y

Enter the data 30

Do you want to attach more nodes [y/n] y

Enter the data 60

Do you want to attach more nodes [y/n] y

Enter the data 15

Do you want to attach more nodes [y/n] y

Enter the data 40

Do you want to attach more nodes [y/n] y

Enter the data 55

Do you want to attach more nodes [y/n] y

Enter the data 70

Do you want to attach more nodes [y/n] n

1.Create TBT

2.Inorder TBT

3.Preorder TBT

4.Exit

Enter your choice :2

15 30 40 50 55 60 70

1.Create TBT

2.Inorder TBT

3.Preorder TBT

4.Exit

Enter your choice :3

50 30 15 40 60 55 70

1.Create TBT

2.Inorder TBT

3.Preorder TBT

4.Exit

Enter your choice :4