```
#include <iostream>
using namespace std;
struct student
{
struct student *lchild;
int lbit;
int roll ;
int rbit;
struct student *rchild;
};
void insert(struct student *tree, struct student *p)
{
if(p->roll< tree->roll)
{
if(tree->lbit==0)
{
tree->lchild=p;
tree->lbit=1;
p->lchild=tree->lchild;
p->rchild=tree;
}
else
insert(tree->lchild,p);
}
else
{
if(tree->rbit==0){
tree->rchild=p;
tree->rbit=1;
p->rchild=tree->rchild;
p->lchild=tree;
}
else
insert(tree->rchild,p);
}
}
struct student *create()
{
int ch;

struct student *root=NULL;
struct student *head= new student;
head->lchild=head->rchild=head;
head->lbit=head->rbit=1;
do
{
struct student *first=new student;
first->lchild=first->rchild=NULL;
first->lbit=first->rbit=0;
cout<<"Enter Roll Number: ";
cin>>first->roll;
if(root==NULL)
{
root=first;
root->lchild=root->rchild=head;
head->lchild=root;
}
else
insert(root,first);
cout<<"Add more[1/0]";
cin>>ch;
}while(ch==1);
return(head);
}
void inorder(struct student *r)
{if(r->lbit==1)
inorder(r->lchild);
cout<< "\t"<< r->roll;
if(r->rbit==1)
inorder(r->rchild);
}
int main()
{
struct student *head;
cout<<"\n To create tree\n ";
head=create();
cout<<"\nInorder traversal is:\n";
inorder(head->lchild);
}
```

Output:-

```
To create tree
 Enter Roll Number: 20
Add more[1/0]1
Enter Roll Number: 15
Add more[1/0]1
Enter Roll Number: 19
Add more[1/0]1
Enter Roll Number: 25
Add more[1/0]1
Enter Roll Number: 36
Add more[1/0]1
Enter Roll Number: 50
Add more[1/0]0

Inorder traversal is:
      15    19    20    25    36    50
```