```
/*A book consists of chapters, chapters consist of sections and sections consist of subsections.
Construct a tree and print the nodes. Find the time and space requirements of your method.
(python/C++).*/
#include<iostream>
#include<stdlib.h>
#include<string.h>
using namespace std;
struct node
{   char name[20];
    node *next;
    node *down;
     int flag;
};
class Gll
{   char ch[20];    int n,i;
    node *head=NULL,*temp=NULL,*t1=NULL,*t2=NULL;
    public:
    node *create();
    void insertb();
    void insertc();
    void inserts();
    void insertss();
    void displayb();

};
```

```
node *Gll::create()
{
   node *p=new(struct node);
   p->next=NULL;
   p->down=NULL;
   p->flag=0;
   cout<<"\n enter the name";
   cin>>p->name;
   return p;
}
void Gll::insertb()
{

       if(head==NULL)
        {   t1=create();
            head=t1;
        }
        else
        {
          cout<<"\n book exist";
        }
}
void Gll::insertc()
{
     if(head==NULL)
```

```
        {
              cout<<"\n there is no book";
         }
         else
         {    cout<<"\n how many chapters you want to insert";
             cin>>n;
             for(i=0;i<n;i++)
             {
             t1=create();
             if(head->flag==0)
             {  head->down=t1;   head->flag=1;    }
             else
             {   temp=head;
                 temp=temp->down;
                while(temp->next!=NULL)
                    temp=temp->next;
                 temp->next=t1;
             }
             }
        }


}
void Gll::inserts()
{
```

```
if(head==NULL)
 {
       cout<<"\n there is no book";
 }
 else
 {    cout<<"\n Enter the name of chapter on which  you want to enter the section";
      cin>>ch;

      temp=head;
    if(temp->flag==0)
    {   cout<<"\n their are no chapters on in book";
    }
    else
    {    temp=temp->down;
     while(temp!=NULL)
      {
          if(!strcmp(ch,temp->name))
          {
                cout<<"\n how many sections you want to enter";
                cin>>n;
                for(i=0;i<n;i++)
                {

                      t1=create();
```

```
                        if(temp->flag==0)
                        {     temp->down=t1;


                              temp->flag=1;  cout<<"\n******";
                              t2=temp->down;


                        }
                        else
                        {
                                    cout<<"\n#####";
                                     while(t2->next!=NULL)
                                     {    t2=t2->next;        }
                                          t2->next=t1;
                         }
                   }
                     break;
               }
                    temp=temp->next;
           }
          }
      }
}
void Gll::insertss()
{
```

```
if(head==NULL)
 {
       cout<<"\n there is no book";
 }
 else
 {    cout<<"\n Enter the name of chapter on which  you want to enter the section";
      cin>>ch;

      temp=head;
    if(temp->flag==0)
    {   cout<<"\n their are no chapters on in book";
    }
    else
    {    temp=temp->down;
     while(temp!=NULL)
      {
          if(!strcmp(ch,temp->name))
          {
             cout<<"\n enter name of section in which you want to enter the sub section";
             cin>>ch;

            if(temp->flag==0)
            {   cout<<"\n their are no sections ";   }
             else
             {      temp=temp->down;
```

```
        while(temp!=NULL)
        {
           if(!strcmp(ch,temp->name))
           {
            cout<<"\n how many subsections you want to enter";
              cin>>n;
for(i=0;i<n;i++)
            {

                t1=create();
                   if(temp->flag==0)
                   {     temp->down=t1;

                         temp->flag=1;  cout<<"\n******";
                         t2=temp->down;

                   }
                  else
                   {
                              cout<<"\n#####";
                               while(t2->next!=NULL)
                               {    t2=t2->next;        }
                                    t2->next=t1;
                    }
               }
```

```
                    break;
                }     temp=temp->next;
               }
            }
          }

              temp=temp->next;
        }
       }
     }
}
void Gll::displayb()
{

       if(head==NULL)
       {  cout<<"\n book not exist";
       }
       else
       {
        temp=head;

          cout<<"\n NAME OF BOOK:  "<<temp->name;
             if(temp->flag==1)
             {
             temp=temp->down;
```

```
                while(temp!=NULL)
                {    cout<<"\n\t\tNAME OF CHAPTER:  "<<temp->name;
                    t1=temp;
                    if(t1->flag==1)
                    {  t1=t1->down;
                      while(t1!=NULL)
                      {    cout<<"\n\t\t\t\tNAME OF SECTION:  "<<t1->name;
                           t2=t1;
                           if(t2->flag==1)
                           {  t2=t2->down;
                           while(t2!=NULL)
                           {    cout<<"\n\t\t\t\t\t\tNAME OF SUBSECTION:  "<<t2->name;
                           t2=t2->next;
                           }
                           }
                           t1=t1->next;
                      }
                    }
                     temp=temp->next;
                }

               }
          }
}
```

```
int main()
{   Gll g;   int x;
     while(1)
    {   cout<<"\n\n enter your choice";
        cout<<"\n 1.insert book";
        cout<<"\n 2.insert chapter";
        cout<<"\n 3.insert section";
        cout<<"\n 4.insert subsection";
        cout<<"\n 5.display book";
        cout<<"\n 6.exit";
        cin>>x;
       switch(x)
       {  case 1:        g.insertb();
                           break;
          case 2:        g.insertc();
                           break;
          case 3:        g.inserts();
                           break;
          case 4:        g.insertss();
                           break;
          case 5:        g.displayb();
                           break;
          case 6:  exit(0);
       }
     }
```

```
    return 0;
}
```