Name : Umesh .V. Jadhav

Practical No : 3(B)

Batch : S1

Roll No : 20

Branch : AI & DS

```
/*
#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.
*/

#include <iostream>
#include <cstdlib>
#include <string.h>
using namespace std;

/* Node Declaration */
struct node
{
   char label[10];
   int ch_count;
   struct node *child[10];
}*root;

/* Class Declaration */
class GT
{
   public:
      void create_tree();
      void display(node *r1);
```

```
    GT()
    {
       root = NULL;
    }
};

void GT::create_tree()
{
   int tchapters, i, j, k;
   root = new node;
   cout << "Enter name of book: ";
   cin >> root->label;
   cout << "Enter number of chapters in book: ";
   cin >> tchapters;
   root->ch_count = tchapters;

   for(i = 0; i < tchapters; i++)
   {
      root->child[i] = new node;
      cout << "Enter Chapter " << i+1 << " name: ";
      cin >> root->child[i]->label;
      cout << "Enter number of sections in Chapter " << root->child[i]->label << ": ";
      cin >> root->child[i]->ch_count;

      for(j = 0; j < root->child[i]->ch_count; j++)
      {
         root->child[i]->child[j] = new node;
         cout << "Enter Section " << j+1 << " name: ";
         cin >> root->child[i]->child[j]->label;
         // You can further ask for subsections if necessary
      }
```

```
    }
}

void GT::display(node *r1)
{
    int i, j, tchapters;
    if(r1 != NULL)
    {
        cout << "\n-----Book Hierarchy---";
        cout << "\nBook title: " << r1->label;
        tchapters = r1->ch_count;

        for(i = 0; i < tchapters; i++)
        {
            cout << "\n  Chapter " << i+1 << ": " << r1->child[i]->label;
            cout << "\n Sections:";
            for(j = 0; j < r1->child[i]->ch_count; j++)
            {
                cout << "\n   " << r1->child[i]->child[j]->label;
            }
        }
    }
}

/* Main Contains Menu */
int main()
{
    int choice;
    GT gt;

    while (1)
```

```
    {
        cout << "-----------------" << endl;
        cout << "Book Tree Creation" << endl;
        cout << "-----------------" << endl;
        cout << "1. Create" << endl;
        cout << "2. Display" << endl;
        cout << "3. Quit" << endl;
        cout << "Enter your choice: ";
        cin >> choice;

        switch(choice)
        {
        case 1:
            gt.create_tree();
            break;  // Add break here to avoid falling through
        case 2:
            gt.display(root);
            break;
        case 3:
            exit(1);
        default:
            cout << "Wrong choice" << endl;
        }
    }
}
```

Output :

(base) computer@computer-ThinkCentre-neo-50s-Gen-3:~$ cd ..

(base) computer@computer-ThinkCentre-neo-50s-Gen-3:/home$ cd computer/

(base) computer@computer-ThinkCentre-neo-50s-Gen-3:~$ g++ Program3.cpp

(base) computer@computer-ThinkCentre-neo-50s-Gen-3:~$ ./a.out

---

Book Tree Creation

---

1. Create

2. Display

3. Quit

Enter your choice: 1

Enter name of book: Programming Basics

Enter number of chapters in book: 2

Enter Chapter 1 name: Introduction

Enter number of sections in Chapter Introduction: 2

Enter Section 1 name: History

Enter Section 2 name: Overview

Enter Chapter 2 name: Advanced Topics

Enter number of sections in Chapter Advanced Topics: 1

Enter Section 1 name: Algorithms

Book Tree Creation

---

1. Create

2. Display

3. Quit

Enter your choice: 2

-----Book Hierarchy---

Book title: Programming Basics

Chapter 1: Introduction

Sections:

History

Overview

Chapter 2: Advanced Topics

Sections:

Algorithms

Book Tree Creation

---

1. Create

2. Display

3. Quit

Enter your choice: 3

(base) computer@computer-ThinkCentre-neo-50s-Gen-3:~$