Assignment 3

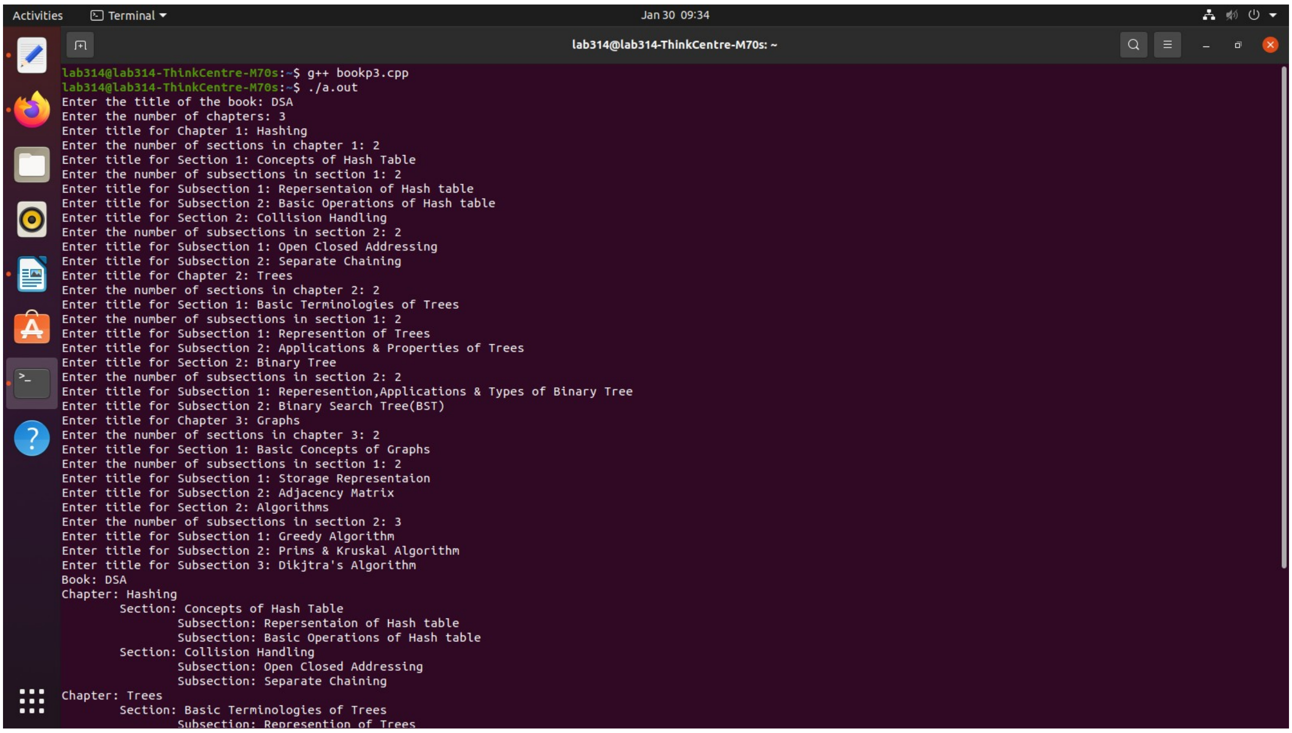
Problem Statement:-

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.

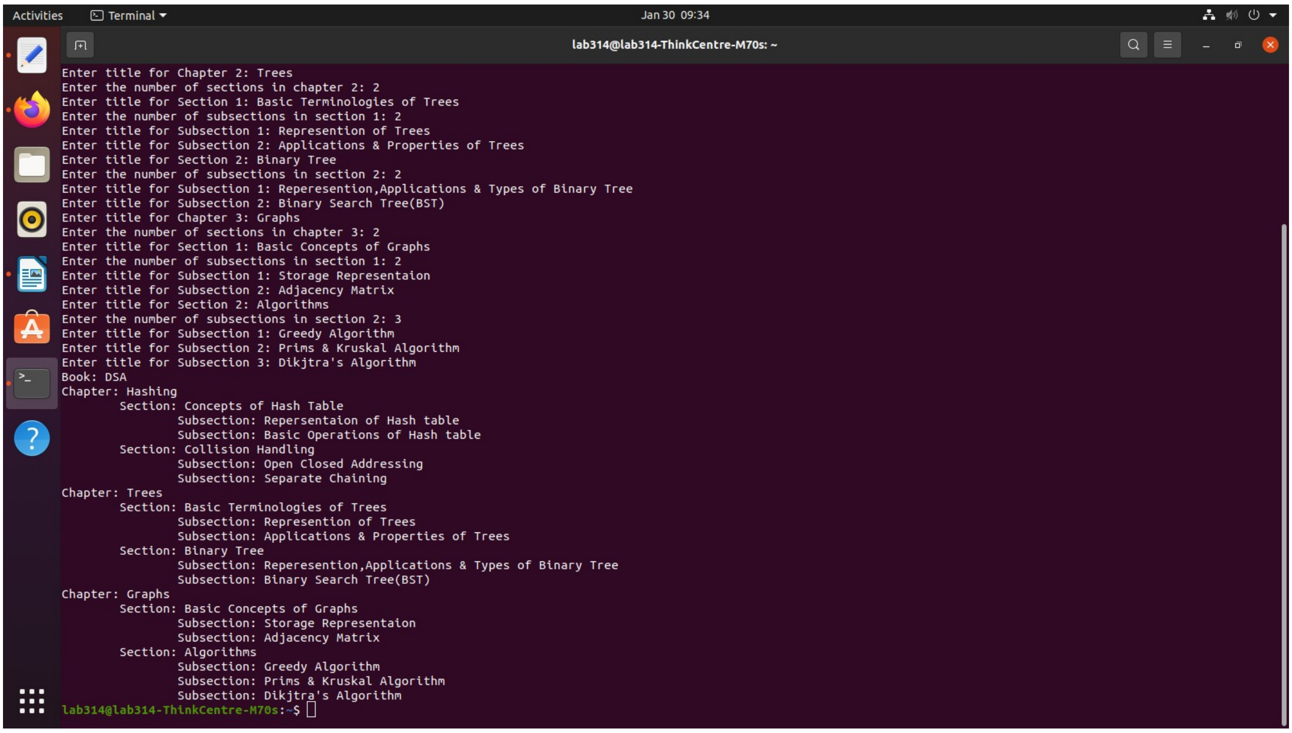
Source Code:-

```
#include <iostream>
#include <vector>
#include <string>

using namespace std;

// Class to represent a Subsection
class Subsection {
public:
    string name;
    Subsection(string name) {
        this->name = name;
    }
};

// Class to represent a Section
class Section {
public:
    string name;
    vector<Subsection*> subsections;

    Section(string name) {
        this->name = name;
    }

    // Add a Subsection to the section
    void addSubsection(Subsection* subsection) {
        subsections.push_back(subsection);
    }

    // Print subsections
    void printSubsections() {
        for (auto& subsection : subsections) {
            cout << "\t\tSubsection: " << subsection->name << endl;
        }
    }
};

// Class to represent a Chapter
class Chapter {
public:
    string name;
    vector<Section*> sections;

    Chapter(string name) {
        this->name = name;
    }

    // Add a Section to the chapter
    void addSection(Section* section) {
        sections.push_back(section);
    }

    // Print sections
    void printSections() {
        for (auto& section : sections) {
            cout << "\tSection: " << section->name << endl;
            section->printSubsections(); // Print subsections under this section
        }
    }
};

// Class to represent a Book
class Book {
public:
    string title;
    vector<Chapter*> chapters;

    Book(string title) {
        this->title = title;
    }

    // Add a Chapter to the book
    void addChapter(Chapter* chapter) {
        chapters.push_back(chapter);
    }

    // Print the book structure
    void printBook() {
        cout << "Book: " << title << endl;
        for (auto& chapter : chapters) {
            cout << "Chapter: " << chapter->name << endl;
            chapter->printSections(); // Print sections and subsections for each chapter
        }
    }
};

int main() {
    string bookTitle, chapterTitle, sectionTitle, subsectionTitle;
    int numChapters, numSections, numSubsections;

    // Taking input for the Book
    cout << "Enter the title of the book: ";
    getline(cin, bookTitle);

    Book* book = new Book(bookTitle);

    // Taking input for Chapters
    cout << "Enter the number of chapters: ";
    cin >> numChapters;
    cin.ignore(); // To ignore the newline character after the number input

    for (int i = 0; i < numChapters; ++i) {
        cout << "Enter title for Chapter " << (i + 1) << ": ";
        getline(cin, chapterTitle);

        Chapter* chapter = new Chapter(chapterTitle);

        // Taking input for Sections in the Chapter
        cout << "Enter the number of sections in chapter " << (i + 1) << ": ";
        cin >> numSections;
        cin.ignore();

        for (int j = 0; j < numSections; ++j) {
            cout << "Enter title for Section " << (j + 1) << ": ";
            getline(cin, sectionTitle);

            Section* section = new Section(sectionTitle);

            // Taking input for Subsections in the Section
            cout << "Enter the number of subsections in section " << (j + 1) << ": ";
            cin >> numSubsections;
            cin.ignore();
            for (int k = 0; k < numSubsections; ++k) {
                cout << "Enter title for Subsection " << (k + 1) << ": ";
                getline(cin, subsectionTitle);

                Subsection* subsection = new Subsection(subsectionTitle);
                section->addSubsection(subsection);
            }

            chapter->addSection(section);
        }

        book->addChapter(chapter);
    }

    // Print the book structure
    book->printBook();

    // Cleanup memory
    delete book;

    return 0;
}
```

Output:-



```
lab314@lab314-ThinkCentre-M70s:~$ g++ bookp3.cpp
lab314@lab314-ThinkCentre-M70s:~$ ./a.out
Enter the title of the book: DSA
Enter the number of chapters: 3
Enter title for Chapter 1: Hashing
Enter the number of sections in chapter 1: 2
Enter title for Section 1: Concepts of Hash Table
Enter the number of subsections in section 1: 2
Enter title for Subsection 1: Repersentaion of Hash table
Enter title for Subsection 2: Basic Operations of Hash table
Enter title for Section 2: Collision Handling
Enter the number of subsections in section 2: 2
Enter title for Subsection 1: Open Closed Addressing
Enter title for Subsection 2: Separate Chaining
Enter title for Chapter 2: Trees
Enter the number of sections in chapter 2: 2
Enter title for Section 1: Basic Terminologies of Trees
Enter the number of subsections in section 1: 2
Enter title for Subsection 1: Represention of Trees
Enter title for Subsection 2: Applications & Properties of Trees
Enter title for Section 2: Binary Tree
Enter the number of subsections in section 2: 2
Enter title for Subsection 1: Reperesention,Applications & Types of Binary Tree
Enter title for Subsection 2: Binary Search Tree(BST)
Enter title for Chapter 3: Graphs
Enter the number of sections in chapter 3: 2
Enter title for Section 1: Basic Concepts of Graphs
Enter the number of subsections in section 1: 2
Enter title for Subsection 1: Storage Representaion
Enter title for Subsection 2: Adjacency Matrix
Enter title for Section 2: Algorithms
Enter the number of subsections in section 2: 3
Enter title for Subsection 1: Greedy Algorithm
Enter title for Subsection 2: Prims & Kruskal Algorithm
Enter title for Subsection 3: Dikjtra's Algorithm
Book: DSA
Chapter: Hashing
        Section: Concepts of Hash Table
                Subsection: Repersentaion of Hash table
                Subsection: Basic Operations of Hash table
        Section: Collision Handling
                Subsection: Open Closed Addressing
                Subsection: Separate Chaining
Chapter: Trees
        Section: Basic Terminologies of Trees
                Subsection: Represention of Trees
```



```
Enter title for Chapter 2: Trees
Enter the number of sections in chapter 2: 2
Enter title for Section 1: Basic Terminologies of Trees
Enter the number of subsections in section 1: 2
Enter title for Subsection 1: Represention of Trees
Enter title for Subsection 2: Applications & Properties of Trees
Enter title for Section 2: Binary Tree
Enter the number of subsections in section 2: 2
Enter title for Subsection 1: Reperesention,Applications & Types of Binary Tree
Enter title for Subsection 2: Binary Search Tree(BST)
Enter title for Chapter 3: Graphs
Enter the number of sections in chapter 3: 2
Enter title for Section 1: Basic Concepts of Graphs
Enter the number of subsections in section 1: 2
Enter title for Subsection 1: Storage Representaion
Enter title for Subsection 2: Adjacency Matrix
Enter title for Section 2: Algorithms
Enter the number of subsections in section 2: 3
Enter title for Subsection 1: Greedy Algorithm
Enter title for Subsection 2: Prims & Kruskal Algorithm
Enter title for Subsection 3: Dikjtra's Algorithm
Book: DSA
Chapter: Hashing
        Section: Concepts of Hash Table
                Subsection: Repersentaion of Hash table
                Subsection: Basic Operations of Hash table
        Section: Collision Handling
                Subsection: Open Closed Addressing
                Subsection: Separate Chaining
Chapter: Trees
        Section: Basic Terminologies of Trees
                Subsection: Represention of Trees
                Subsection: Applications & Properties of Trees
        Section: Binary Tree
                Subsection: Reperesention,Applications & Types of Binary Tree
                Subsection: Binary Search Tree(BST)
Chapter: Graphs
        Section: Basic Concepts of Graphs
                Subsection: Storage Representaion
                Subsection: Adjacency Matrix
        Section: Algorithms
                Subsection: Greedy Algorithm
                Subsection: Prims & Kruskal Algorithm
                Subsection: Dikjtra's Algorithm
lab314@lab314-ThinkCentre-M70s:~$ 
```