Assignment no: 1(B)

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.

Objective:

To understand the basic concept of Non Linear Data Structure and its basic operation in Data structure.

Outcome:

To implement the basic concept of Tree to store a numbers in it. Also perform basic Operation Insert, Delete and search in in tree in Data structure.

Software & Hardware Requirements:

1. 64-bit Open source Linux or its derivative
2. Open Source C++ Programming tool like G++/GCC

Theory:

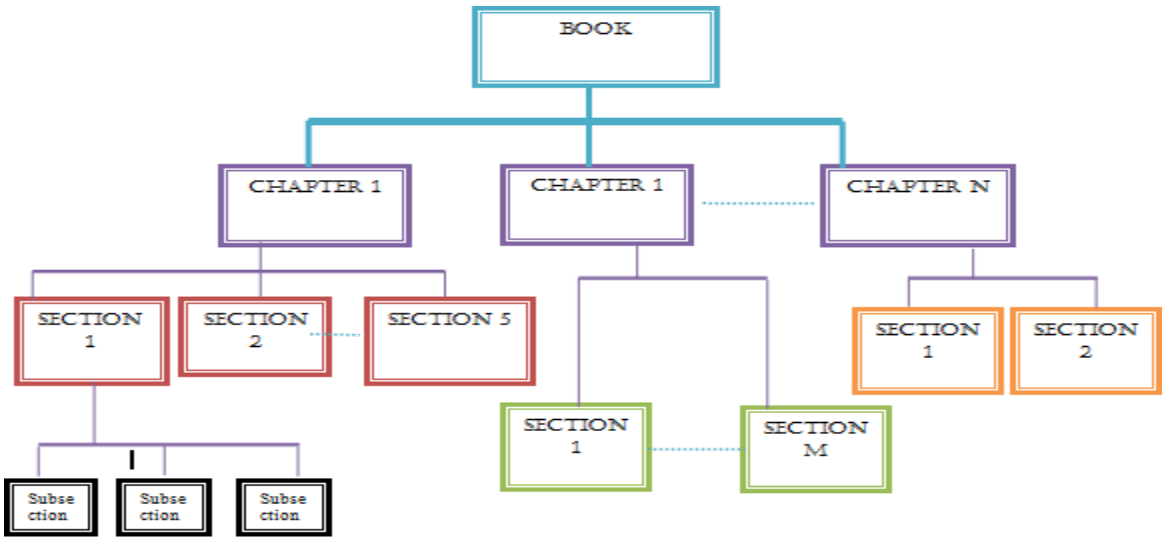
A tree is the non linear data structure represents hierarchal relationship among object (data). The term "tree" was coined in 1857 by the British mathematician Arthur Cayley

Definitions

A **tree** is an undirected simple graph *G* that satisfies any of the following equivalent conditions:
• *G* is connected and has no cycles.
• *G* has no cycles, and a simple cycle is formed if any edge is added to *G*.
• *G* is connected, but is not connected if any single edge is removed from *G*.
If *G* has finitely many vertices, say *n* of them, then the above statements are also equivalent to any of the following conditions:
• *G* is connected and has $n - 1$ edges.

Tree represents the nodes connected by edges.

Binary Tree is a special data structure used for data storage purposes. A binary tree has a special condition that each node can have a maximum of two children. A binary tree has the benefits of both an ordered array and a linked list as search is as quick as in a sorted array and insertion or deletion operation are as fast as in linked list.



TREE : HIERARCHICAL STRUCTURE OF BOOK, CHAPTER, SECTION, SUBSECTION

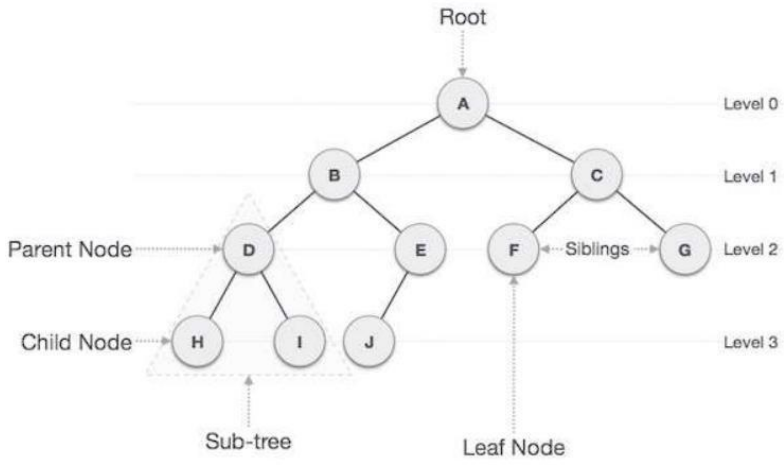
Important Terms

Following are the important terms with respect to tree.

- Path− Path refers to the sequence of nodes along the edges of a tree.
- Root–The node at the top of the tree is called root. There is only one root per tree and one path from the root node to any node.
- Parent− Any node except the root node has one edge upward to a node called parent.
- Child–The node below a given node connected by its edge downward is called its child node.
- Leaf –The node which does not have any child node is called the leaf node.
- Subtree− Subtree represents the descendants of a node

Visiting − Visiting refers to checking the value of a node when control is on the node.

- Traversing− Traversing means passing through nodes in a specific order.
- Levels− Level of a node represents the generation of a node. If the root node is at level 0, then its next child node is at level 1, its grandchild is at level 2,and so on.
- Keys− Key represents a value of a node based on which a search operation is to be carried out for a node.



Applications of trees

- Class hierarchy in Java
- file system
- storing hierarchies in organizations

Conclusion: we study about the non li