Trees

Binary Trees

The notion of a tree is an important concept in computing. Trees represent hierarchical relationships among elements of a set.

We begin with some definitions.

Tree

1. A single node is a tree and this node is the root of the tree.
2. Suppose r is a node and $T_1, T_2, \ldots, T_k$ are trees with roots $r_1, r_2, \ldots, r_k$, respectively, then we can construct a new tree whose root is r and $T_1, T_2, \ldots, T_k$ are the subtrees of the root. The nodes $r_1, r_2, \ldots, r_k$ are called the children of r.

A tree with root r and the subtrees $T_1, T_2, \ldots, T_k$ is shown in Figure 1.

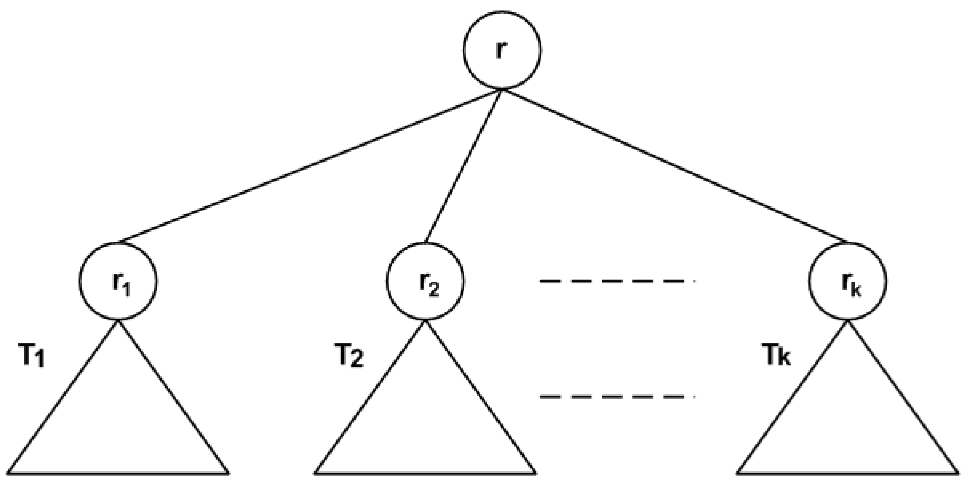

Figure 1: Recursive structure of a tree

A path in a tree is a sequence of nodes $n_1, n_2, \ldots, n_k$, such that n_i is the parent of n_i+1 for i = 1, 2, . . . , k–1. The length of a path is 1 less than the number of nodes on the path. Thus there is a path of length zero from a node to itself.

If there is a path from node x to node y, then
  x is called an ancestor of y and
  y is called a descendent of x.

If x /= y, then x is a proper ancestor of y and y is a proper descendent of x.

A subtree of a tree is a node in the tree together with all its descendants.

A tree node that has no proper descendants is called a *leaf* node.

The *height* of a node in a tree is the length of a longest path from the node to a leaf.
The *height* of a tree is the height of its root.
The *depth* of a node is the length of the unique path from the root to the node.

Tree Traversals

A systematic way of ordering the nodes of a tree is sometimes called a tree traversal. There are three well known tree traversal, namely the *Preorder*, *Inorder*, and *Postorder*. Each of these traversals constitute a recursive method of listing (or visiting) the nodes (each node exactly once) of the tree.

Preorder, Inorder, Postorder Traversals

- If a tree T consists of a single node, that node itself is the preorder, inorder, and postorder listing of T .
- For any tree T with more than one node
    - The preorder listing of T is the root of T, followed by the nodes of T_1 in preorder, ..., and the nodes of T_k in preorder.
    - The inorder listing of T is the nodes of T_1 in inorder, followed by the root r, followed by the nodes of T_2 in inorder, ..., and the nodes of T_k in inorder.
    - The postorder listing of T is the nodes of T_1 in postorder, ..., the nodes of T_k in postorder, all followed by the root r.

As an example of the Preorder, Inorder, Postorder traversals consider the tree in Figure 2.

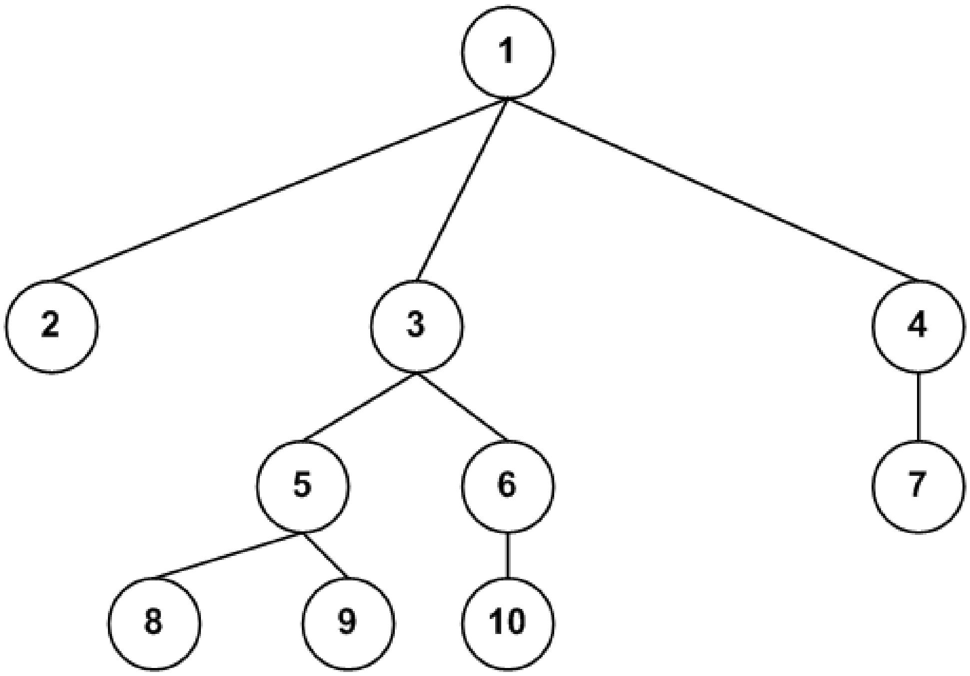

Figure 2: Example of a general tree

The Preorder, Inorder, and Postorder Listings of the tree in Figure 2 are as follows:

- Preorder: 1,2,3,5,8,9,6,10,4,7
- Postorder: 2,8,9,5,10,6,3,7,4,1
- Inorder: 2,1,8,5,9,3,10,6,7,4

Binary Trees

A binary tree is a special kind of tree in which each node has either

- no children, or
- a left child, or
- a right child, or
- both a left and a right child.

Thus each node in a binary tree has at most two children. Moreover, the children are distinguishable as the left or the right child. A conventional way of drawing binary trees is shown in Figure 3. Figure 3 shows several examples of binary trees.

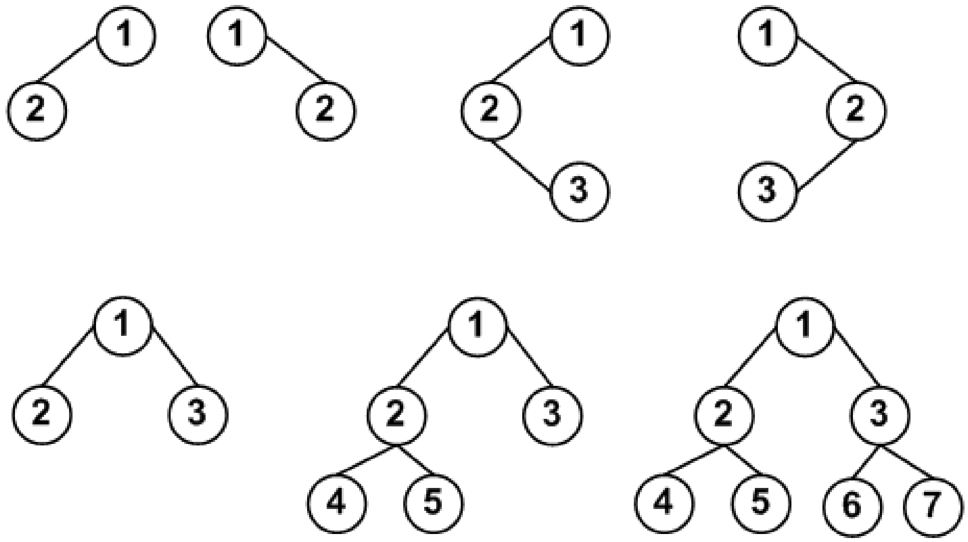

Figure 3: Examples of binary trees

Binary Tree Traversals

- Preorder
  - Visit root, visit left subtree in preorder, visit right subtree in preorder.
- Postorder
  - Visit left subtree in postorder, right subtree in postorder, then the root.
- Inorder
  - Visit left subtree in inorder, then the root, then the right subtree in inorder.

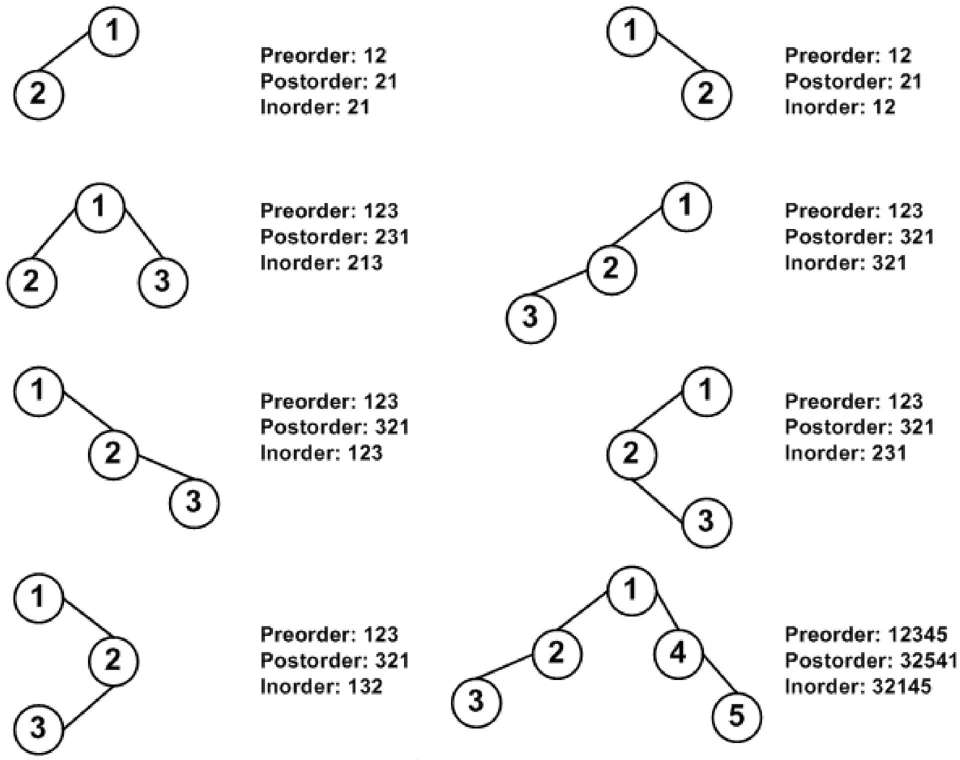

Figure 4: Binary tree traversals

Figure 4 illustrates several binary trees and the preorder, inorder, and postorder traversals of these trees.

Full and Complete binary trees

Since each node has at most two children, it follows that for any binary tree with height h has at most $2h$ leaves. Moreover, the total number of nodes in a binary tree of height h is no more than $2^{h+1} - 1$.

A binary tree with height h and $2^{h+1} - 1$ nodes (or 2^h leaves) is called a full binary tree. Recall that the depth of a node in a tree is the length of the unique path from the root to the node. Clearly, every leaf in a full binary tree has depth h and every node that is not a leaf has both the left and right children.

The nodes of a full binary tree can be numbered in a natural way, level by level, left to right. For example, see the full binary tree in the Figure 3.

Consider a full binary tree T having n or more nodes, and whose nodes are labelled (numbered) as above. A binary tree with n nodes is called a complete binary tree if it contains all the first n nodes of T. Figure 5 shows examples of complete and incomplete binary trees.

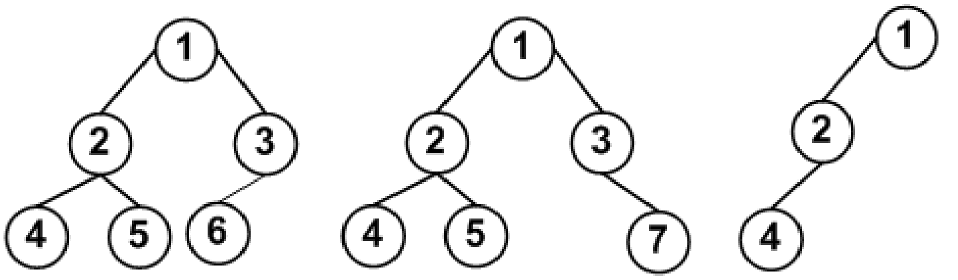

Figure 5: Examples of complete, incomplete binary trees

Note that for any node i of a complete binary tree the following holds:

- The left child of node i, if it exists, is node $2i$.
- The right child of node i, if it exists, is node $2i + 1$.
- The parent of node i is node $[i/2]$