COMP 352 – FALL 2021

Tutorial 6

Session Outline

- Binary Trees:
  - Definitions
  - Additional Methods
  - Properties and Implementations
  - Additional Traversal Algorithms
    - In-order Traversal
    - Euler Tour Traversal
- Problem Solving

Binary Trees - Definitions

A ***Binary Tree*** is an ordered tree having the following properties:

1. Every node has at most 2 children
2. Each child node is labeled as being either a ***left child*** or a ***right child.***
3. A left child precedes a right child in the ordering of children of a node.

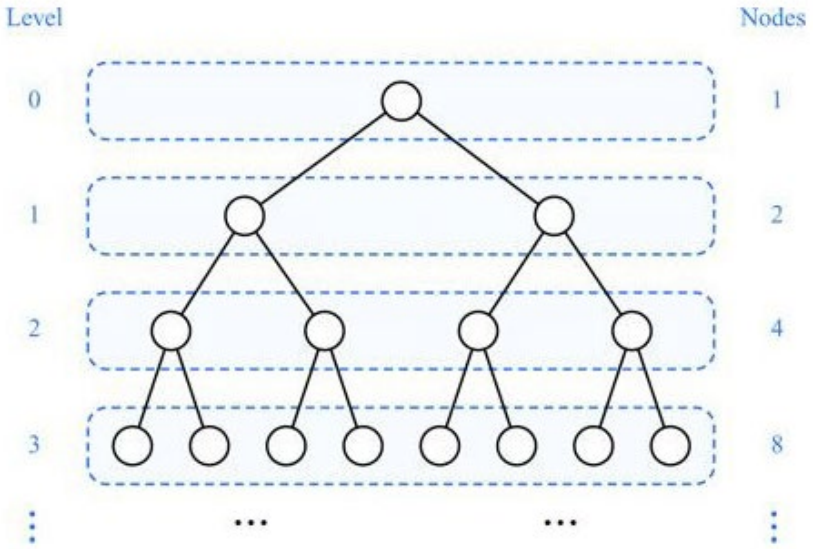
A binary tree is ***proper*** (also referred to as **full** binary tree) if each node has either zero or two children. Thus, in a proper binary tree, every internal node has exactly two children. A binary tree that is not proper is ***improper***.

Binary Trees - Additional Methods

A ***Binary Tree*** supports the following 4 accessor methods in addition to the regular Tree methods:

- left(v): Return the left child of v; an error condition occurs if v has no left child.
- right(v): Return the right child of v; an error condition occurs if v has no right child.
- hasLeft(v): Test whether v has a left child.
- hasRight(v): Test whether v has a right child.

Binary Trees – Properties & Implementations



In a ***Binary Tree***, every level d has at most 2^d nodes (the rest of the properties on p.303)

Linked structure implementation & performance (p.305)

Array list representation (p.314)

Binary Trees - Additional Traversal Algorithms

***Binary Tree*-specific traversal algorithms:**

- The inorder traversal, informally viewed as visiting the nodes from left to right.
- The Euler Tour Traversal, informally described as a "walk around" the tree, where every node is traversed 3 times (on the left, from below, and on the right)

Problem Solving – Question 1

Draw an arithmetic expression tree that has four external nodes, storing the numbers 1, 5, 6, and 7 (with each number stored in a distinct external node, but not necessarily in this order), and has three internal nodes, each storing an operator from the set { + , − , ×, /}, so that the value of the root is 21. The operators may return and act on fractions, and an operator may be used more than once.

Problem Solving – Question 2

Draw a (single) binary tree T such that:

- Each internal node of T stores a single character
- A *preorder* traversal of T yields EXAMFUN
- An inorder traversal of T yields MAFXUEN.

Problem Solving – Question 3

- Draw a Binary Search Tree that shows the result of the tree after inserting the following keys (from left to right): The tree initially is empty.

  key ={17, 9, 26, 12, 11, 7, 30, 20, 21, 10}

Problem Solving – Question 4

Describe, in pseudo-code, a nonrecursive method for performing an in-order traversal of a binary tree in linear time.