SECTION 10.6

Rooted Trees

Rooted Trees

An outdoor tree is rooted and so is the kind of family tree that shows all the descendants of one particular person. The terminology and notation of rooted trees blends the language of botanical trees and that of family trees.

In mathematics, a **rooted tree** is a tree in which one vertex has been distinguished from the others and is designated the ***root***.

Given any other vertex *v* in the tree, there is a unique path from the root to *v*. (After all, if there were two distinct paths, a circuit could be constructed.)

Rooted Trees

The number of edges in such a path is called the *level* of v, *i.e.*, the distance of v from the root.

The *height* of the tree is the length of the longest such path.

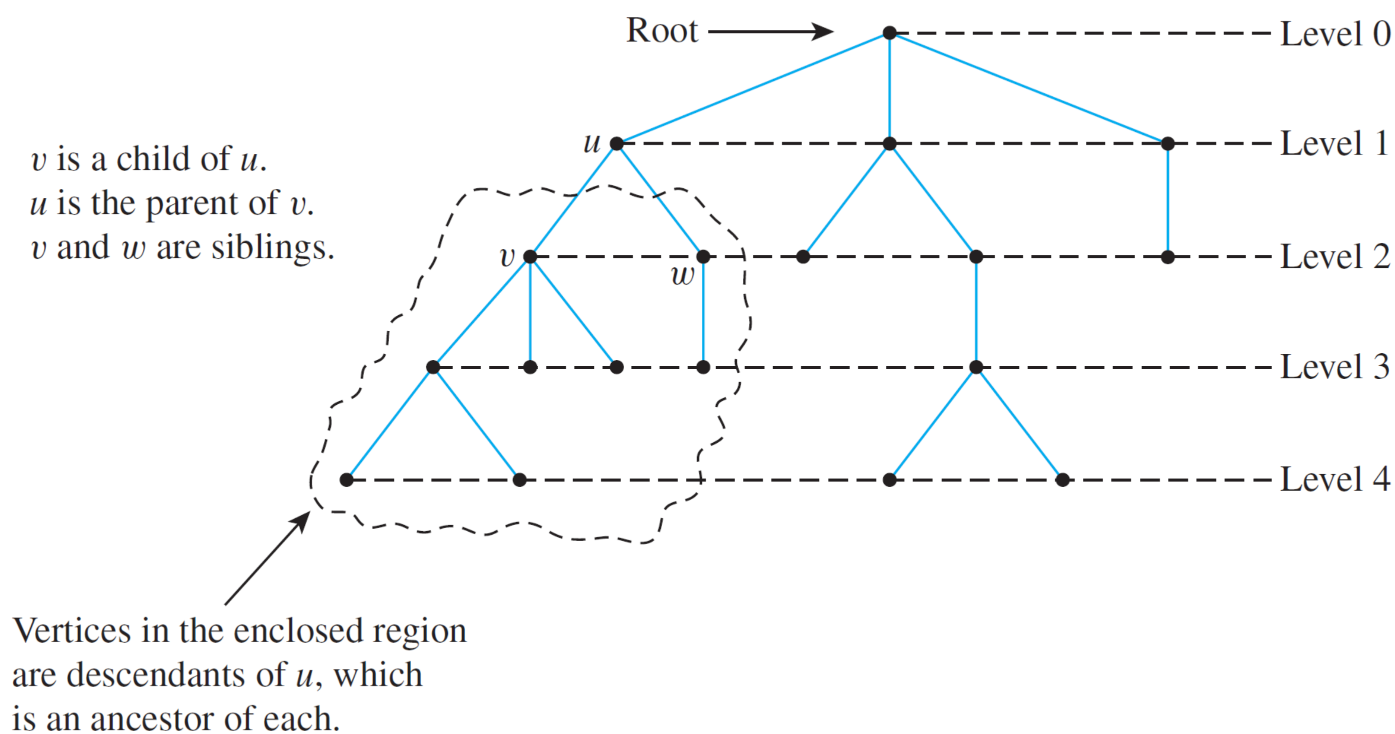
It is traditional in drawing rooted trees to place the root at the top and show the branches descending from it.

• Definition

A **rooted tree** is a tree in which there is one vertex that is distinguished from the others and is called the **root.** The **level** of a vertex is the number of edges along the unique path between it and the root. The **height** of a rooted tree is the maximum level of any vertex of the tree. Given the root or any internal vertex v of a rooted tree, the **children** of v are all those vertices that are adjacent to v and are one level farther away from the root than v. If w is a child of v, then v is called the **parent** of w, and two distinct vertices that are both children of the same parent are called **siblings.** Given two distinct vertices v and w, if v lies on the unique path between w and the root, then v is an **ancestor** of w and w is a **descendant** of v.

Rooted Trees

These terms are illustrated in Figure 10.6.1.



A Rooted Tree

Figure 10.6.1

Example 1 – *Rooted Trees*

Consider the tree with root v_0 shown on the right.

a. What is the level of v_5?

b. What is the level of v_0?

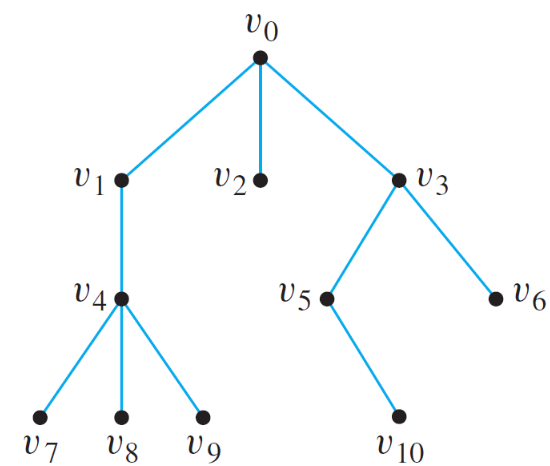
c. What is the height of this rooted tree?

d. What are the children of v_3?

e. What is the parent of v_2?

f. What are the siblings of v_8?

g. What are the descendants of v_3?



Example 1 – *Solution*

a. 2

b. 0

c. 3

d. v_5 and v_6

e. v_0

f. v_7 and v_9

g. v_5, v_6, v_{10}

Rooted Trees

Note that in the tree with root v_0 shown below, v_1 has level 1 and is the child of v_0, and both v_0 and v_1 are terminal vertices.

Binary Trees

Binary Trees

When every vertex in a rooted tree has at most two children and each child is designated either the (unique) left child or the (unique) right child, the result is a ***binary tree***.

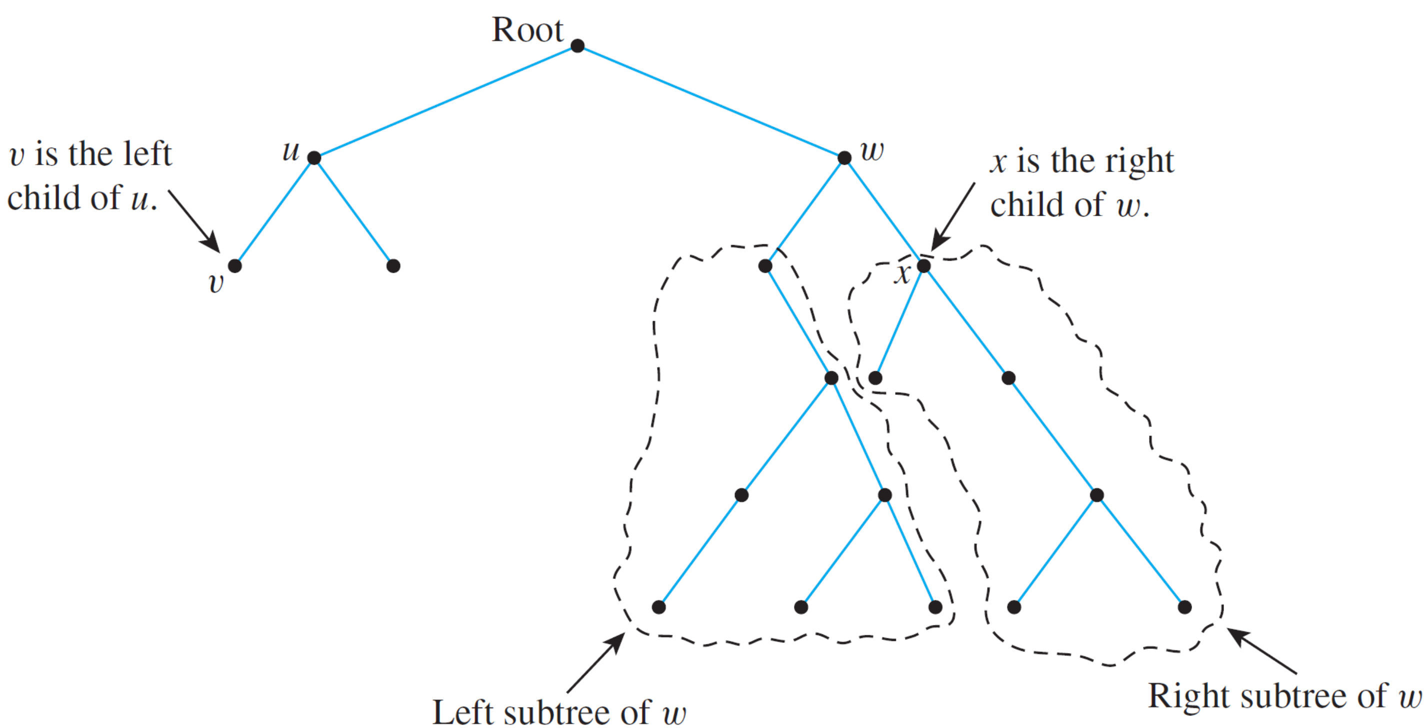
• Definition

A **binary tree** is a rooted tree in which every parent has at most two children. Each child in a binary tree is designated either a **left child** or a **right child** (but not both), and every parent has at most one left child and one right child. A **full binary tree** is a binary tree in which each parent has exactly two children.

Given any parent v in a binary tree T, if v has a left child, then the **left subtree** of v is the binary tree whose root is the left child of v, whose vertices consist of the left child of v and all its descendants, and whose edges consist of all those edges of T that connect the vertices of the left subtree. The **right subtree** of v is defined analogously.

Binary Trees

These terms are illustrated in Figure 10.6.2.



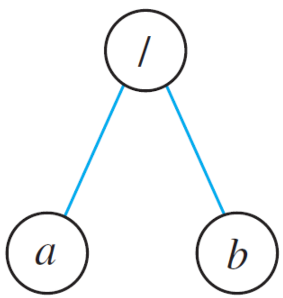
A Binary Tree

Figure 10.6.2

Example 2 – *Representation of Algebraic Expressions*

Binary trees are used in many ways in computer science. One use is to represent algebraic expressions with arbitrary nesting of balanced parentheses.

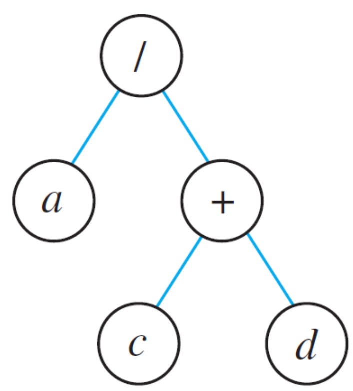
For instance, the following (labeled) binary tree represents the expression *a*/*b*: The operator is at the root and acts on the left and right children of the root in left-right order.



Example 2 – *Representation of Algebraic Expressions*

cont'd

More generally, the binary tree shown below represents the expression $a/(c + d)$. In such a representation, the internal vertices are arithmetic operators, the terminal vertices are variables, and the operator at each vertex acts on its left and right subtrees in left-right order.



Draw a binary tree to represent the expression
$((a - b) \cdot c) + (d/e)$.

Characterizing Binary Trees

An interesting theorem about full binary trees says that if you know the number of internal vertices of a full binary tree, then you can calculate both the total number of vertices and the number of terminal vertices, and conversely.

More specifically, a full binary tree with k internal vertices has a total of $2k + 1$ vertices of which $k + 1$ are terminal vertices.

Theorem 10.6.1

If k is a positive integer and T is a full binary tree with k internal vertices, then T has a total of $2k + 1$ vertices and has $k + 1$ terminal vertices.

Characterizing Binary Trees

Another interesting theorem about binary trees specifies the maximum number of terminal vertices of a binary tree of a given height.

The maximum number of terminal vertices of a binary tree of height h is 2^h. Another way to say this is that a binary tree with t terminal vertices has height of at least $\log_2 t$.

Theorem 10.6.2

For all integers $h \geq 0$, if T is any binary tree with of height h and t terminal vertices, then

$$t \leq 2^h.$$

Equivalently, $$\log_2 t \leq h.$$