Trees

A *tree* is a hierarchy. At the top is the *root*, the root can have *children*, the children can have children, and so on.

*unordered trees*: The order in which siblings are attached to a vertex is arbitrary (just an artifact of the data structure).

*ordered trees*: For example, in a *binary tree*, a vertex can have a right child but no left child.
A tree can be defined recursively as a set of vertices consisting of a root for which no other vertex is designated as its parent, together with a (possibly empty) set $S$ of disjoint trees, where $r$ is the parent of each of the roots of the trees in $S$. In the figure, vertices are labeled $a$ through $i$, $a$ is the root, $a$ is the parent of $b$ and $c$, $b$ is the parent of $d$, $c$ is the parent of $e$, $f$, and $g$, and $f$ is the parent of $h$ and $i$. 
(tree terms continued)

**child**: \( v \) is the parent of \( w \) if and only if \( w \) is a *child* of \( v \).

**internal vertex**: A vertex that is not the root and has at least one child; in the figure, \( b, c, \) and \( f \) are *internal vertices*.

**leaf**: A vertex with no children; in the figure, \( d, e, h, i, \) and \( g \) are *leaves*.

**sibling**: If two vertices are children of the same vertex, then they are *siblings*. In the figure, \( b \) and \( c \) are siblings, \( e, f, \) and \( g \) are siblings, and \( h \) and \( i \) are siblings.

**ancestor**: A vertex \( v \) is an *ancestor* of a vertex \( w \) if \( v = w \) or \( v \) can be reached from \( w \) by following the parent relationship; if \( w \neq v \), then \( w \) is a *proper ancestor* of \( v \).

**descendant**: \( w \) is a *descendant* of \( v \) if and only if \( v \) is an ancestor of \( w \); if \( w \neq v \), then \( w \) is a *proper descendant* of \( v \).

**subtree**: The children of a vertex \( v \) are the roots of the *subtrees* of \( v \). In the figure, removing the root \( a \) leaves two subtrees (one subtree rooted at \( b \) and one subtree rooted at \( c \)).
Tree Dimensions

**DEPTH(v):**

The number of edges on the path from the root to vertex $v$ (the depth of the root is 0).

**HEIGHT(v):**

The number of edges on a longest path from vertex $v$ to a leaf (leaves have height 0).

**MIN-HEIGHT(v):**

The number of edges on a shortest path from vertex $v$ to a leaf (leaves have min-height 0).

**LEVEL(v):**

$\text{LEVEL}(v) = \text{HEIGHT}(root) - \text{DEPTH}(v)$