Trees

Chapter 11

Chapter Summary

- Introduction to Trees
- Applications of Trees (*not currently included in overheads*)
- Tree Traversal
- Spanning Trees
- Minimum Spanning Trees (*not currently included in overheads*)

Introduction to Trees

Section 11.1

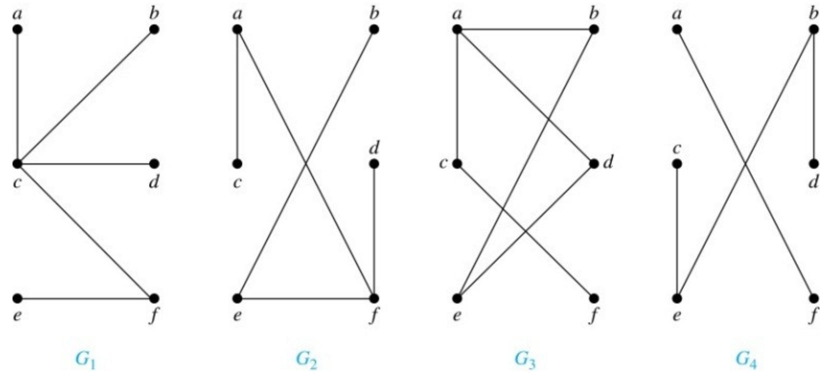
Section Summary

- Introduction to Trees
- Rooted Trees
- Trees as Models
- Properties of Trees

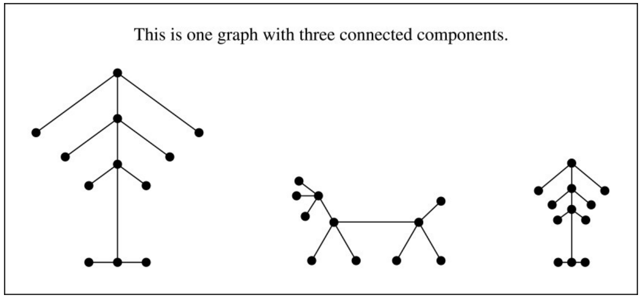
Trees

Definition: A *tree* is a connected undirected graph with no simple circuits.

Example: Which of these graphs are trees?



Definition: A *forest* is a graph that has no simple circuits but is not connected. Each of the connected components in a forest is a tree.



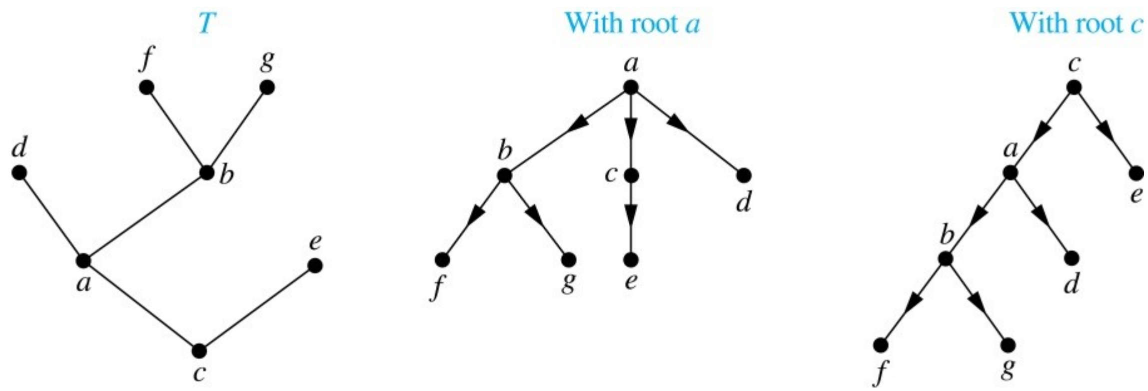
Trees (*continued*)

Theorem: An undirected graph is a tree if and only if there is a unique simple path between any two of its vertices.

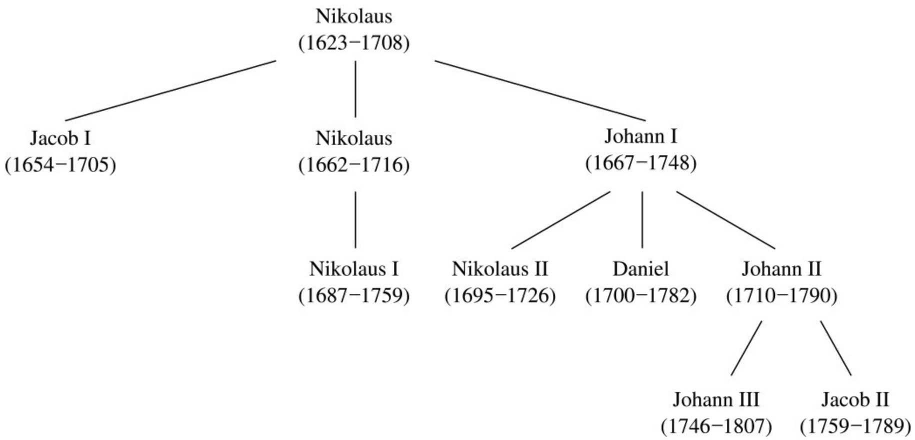
Rooted Trees

Definition: A *rooted tree* is a tree in which one vertex has been designated as the *root* and every edge is directed away from the root.

An unrooted tree is converted into different rooted trees when different vertices are chosen as the root.



Rooted Tree Terminology



- If *v* is a vertex of a rooted tree other than the root, the *parent* of *v* is the unique vertex *u* such that there is a directed edge from *u* to *v*. When *u* is a parent of *v*, *v* is called a *child* of *u*. Vertices with the same parent are called *siblings*.
- The *ancestors* of a vertex are the vertices in the path from the root to this vertex, excluding the vertex itself and including the root. The *descendants* of a vertex *v* are those vertices that have *v* as an ancestor.
- A vertex of a rooted tree with no children is called a *leaf*. Vertices that have children are called *internal vertices*.
- If *a* is a vertex in a tree, the *subtree* with *a* as its root is the subgraph of the tree consisting of *a* and its descendants and all edges incident to these descendants.

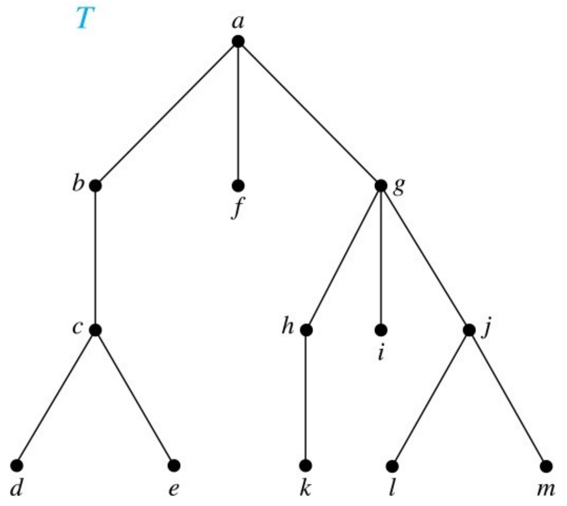
Terminology for Rooted Trees

Example: In the rooted tree *T* (with root *a*):

(i) Find the parent of *c*, the children of *g*, the siblings of *h*, the ancestors of *e*, and the descendants of *b*.

(ii) Find all internal vertices and all leaves.

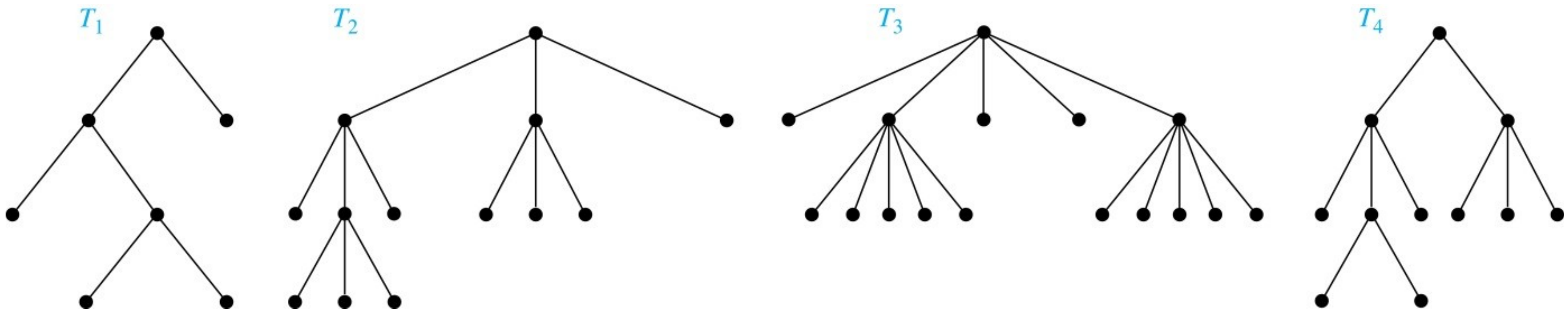
(iii) What is the subtree rooted at *g*?



m-ary Rooted Trees

Definition: A rooted tree is called an *m-ary tree* if every internal vertex has no more than *m* children. The tree is called a *full m-ary tree* if every internal vertex has exactly *m* children. An *m*-ary tree with *m* = 2 is called a *binary* tree.

Example: Are the following rooted trees full *m*-ary trees for some positive integer *m*?



Properties of Trees

Theorem 2: A tree with n vertices has $n - 1$ edges.

***Proof* (*by mathematical induction*):**

BASIS STEP: When $n = 1$, a tree with one vertex has no edges. Hence, the theorem holds when $n = 1$.

INDUCTIVE STEP: Assume that every tree with k vertices has $k - 1$ edges. Suppose that a tree T has $k + 1$ vertices and that v is a leaf of T. Let w be the parent of v. Removing the vertex v and the edge connecting w to v produces a tree T' with k vertices. By the inductive hypothesis, T' has $k - 1$ edges. Because T has one more edge than T', we see that T has k edges. This completes the inductive step. ◀

Counting Vertices in Full *m*-Ary Trees

Theorem 3: A full *m*-ary tree with *i* internal vertices has $n = mi + 1$ vertices.

Proof : Every vertex, except the root, is the child of an internal vertex. Because each of the *i* internal vertices has *m* children, there are *mi* vertices in the tree other than the root. Hence, the tree contains $n = mi + 1$ vertices.

◀

Counting Vertices in Full *m*-Ary Trees (*continued*)

Theorem 4: A full *m*-ary tree with

(*i*) n vertices has $i = (n - 1)/m$ internal vertices and $l = [(m - 1)n + 1]/m$ leaves,

(*ii*) i internal vertices has $n = mi + 1$ vertices and $l = (m - 1)i + 1$ leaves,

(*iii*) l leaves has $n = (ml - 1)/(m - 1)$ vertices and $i = (l - 1)/(m - 1)$ internal vertices.

These all follow from the fact that $n = l + i$ and $n = mi + 1$

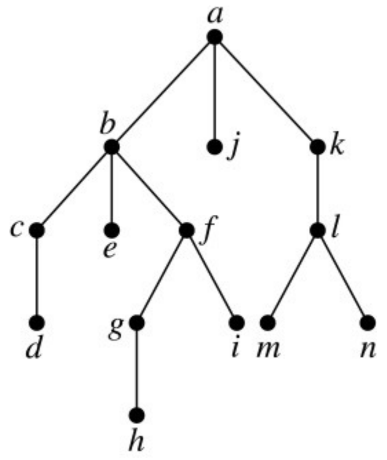
Level of vertices and height of trees

- When working with trees, we often want to have rooted trees where the subtrees at each vertex contain paths of approximately the same length.
- To make this idea precise we need some definitions:
  - The *level* of a vertex *v* in a rooted tree is the length of the unique path from the root to this vertex.
  - The *height* of a rooted tree is the maximum of the levels of the vertices.

Example:

(*i*) Find the level of each vertex in the tree to the right.

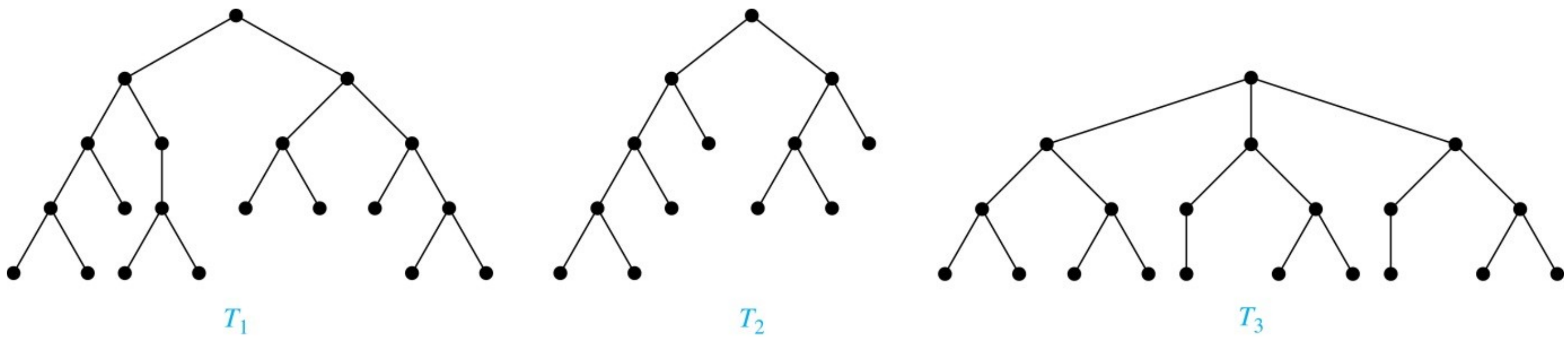
(*ii*) What is the height of the tree?



Balanced *m*-Ary Trees

Definition: A rooted *m*-ary tree of height *h* is *balanced* if all leaves are at levels *h* or *h* − 1.

Example: Which of the rooted trees shown below is balanced?



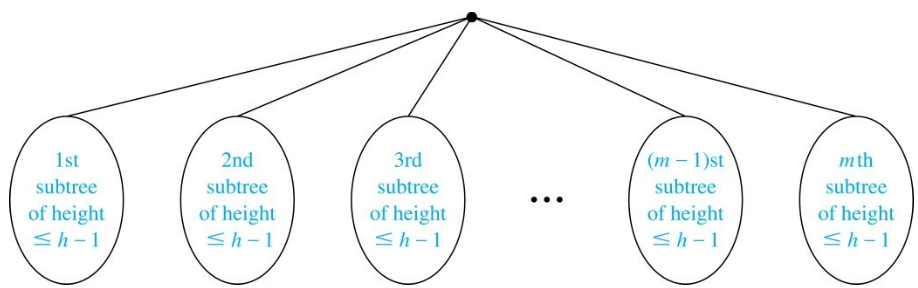
The Bound for the Number of Leaves in an *m*-Ary Tree

Theorem 5: There are at most m^h leaves in an *m*-ary tree of height *h*.

***Proof (by mathematical induction on height)*:**

BASIS STEP: Consider an *m*-ary trees of height 1. The tree consists of a root and no more than *m* children, all leaves. Hence, there are no more than $m^1 = m$ leaves in an *m*-ary tree of height 1.

INDUCTIVE STEP: Assume the result is true for all *m*-ary trees of height < *h*. Let *T* be an *m*-ary tree of height *h*. The leaves of *T* are the leaves of the subtrees of *T* we get when we delete the edges from the root to each of the vertices of level 1.



Each of these subtrees has height $\leq h - 1$. By the inductive hypothesis, each of these subtrees has at most m^{h-1} leaves. Since there are at most *m* such subtrees, there are at most $m \cdot m^{h-1} = m^h$ leaves in the tree.

Corollary 1: If an *m*-ary tree of height *h* has *l* leaves, then $h \geq \lceil \log_m l \rceil$. If the *m*-ary tree is full and balanced, then $h = \lceil \log_m l \rceil$. (*see text for the proof*)