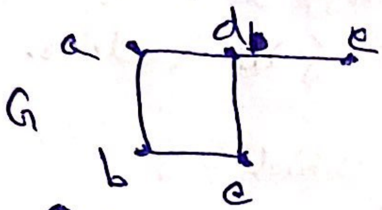
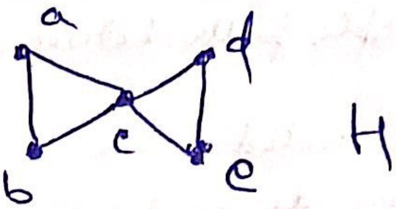
Example



G has no H.C.



H has no H.C.

Example Kn has a H.C. whenever $n \geq 3$

Theorem (Dirac)

If G is a simple graph with n vertices such that deg of every vertex in G is at least $n/2$ then G has a H.C.

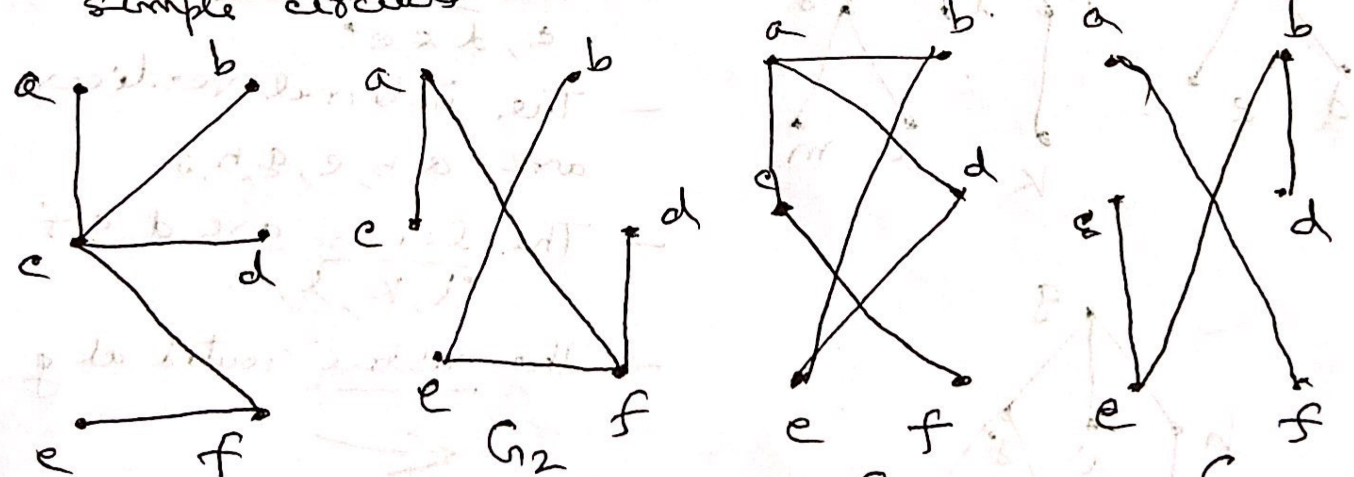
Theorem (ORE)

If G is a simple graph with n vertices ($n \geq 3$) s.t. $\deg(u) + \deg(v) \geq n$ for every pair of nonadjacent vertices u & v in G then G has a H.C.

TREES

Family trees

- A tree is a connected undirected graph with no simple circuits



G1 — $G_1 \to$ tree

G2 — $G_2 \to$ tree

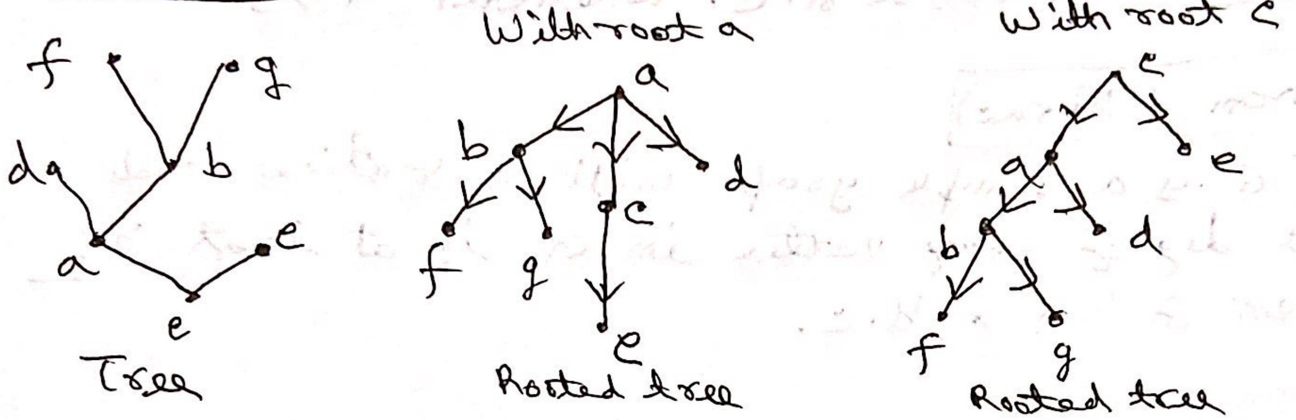
G3 — G3 not tree e, b, a, d, e a circuit

G4 — not tree, not connected

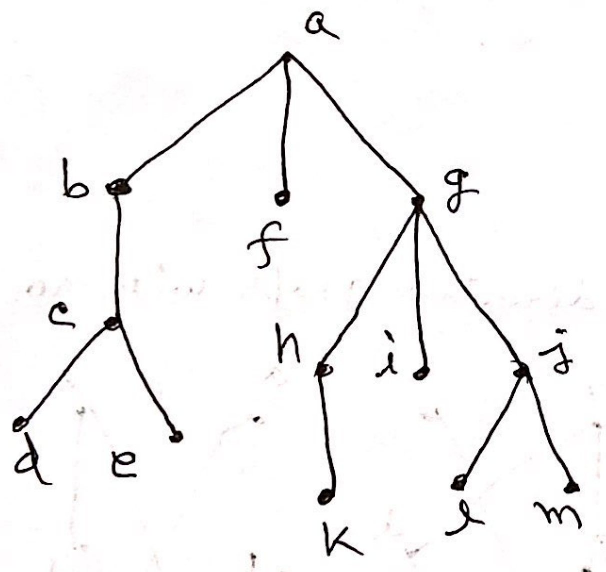
Theorem: An undirected graph is a tree iff there is a unique simple path between any two of its vertices.

Rooted Tree 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.

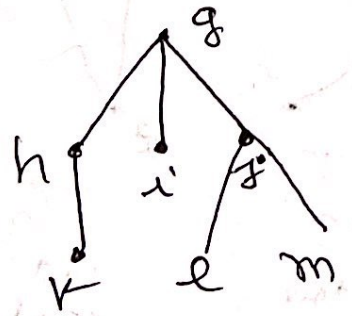
Example



Example



- Parent of 'e' is 'b'
- The children of 'g' are h, i, j
- The ancestors of 'e' are c, b & a
- The descendants of 'b' are c, d & e.
- The internal vertices are a, b, c, g, h, j
- The leaves are d, e, f, i, k, l, m
- The subtree rooted at g

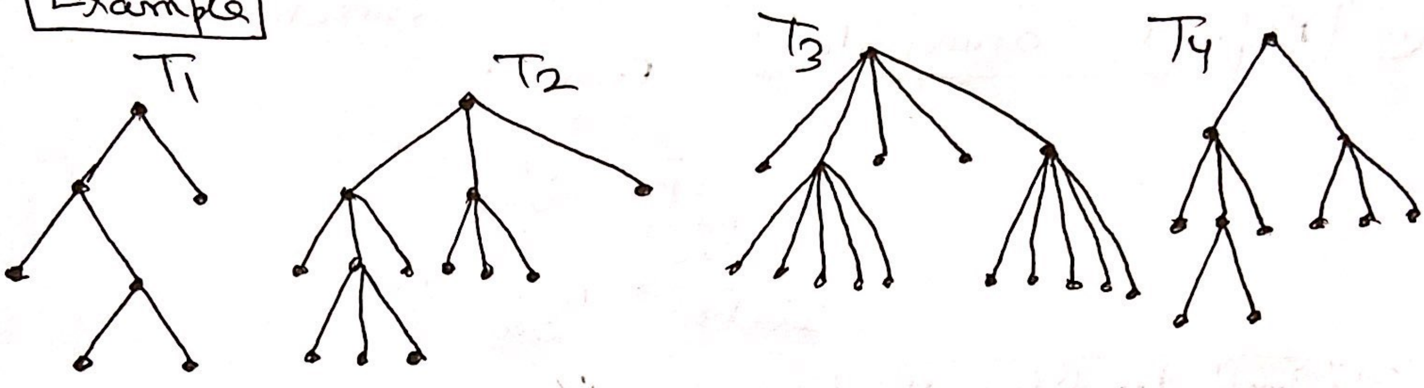


— A rooted tree is called an m-ary tree if every internal vertex has no more than m children.

— 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



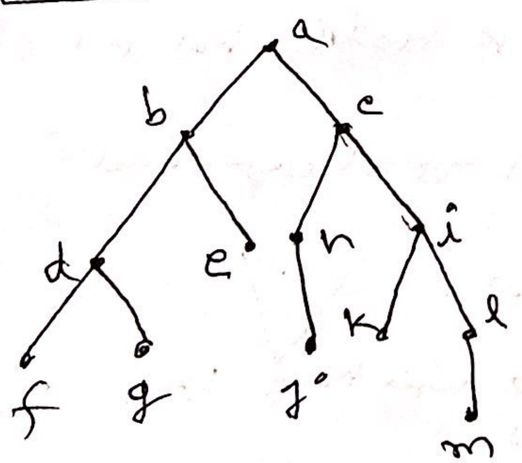
— T_1 is full binary tree because each of its internal vertices has two children.

— T_2 is a full 3-ary tree (∵ each internal vertices has 3 children)

— T_3 is a full 5-ary tree (every internal vertex has 5 children)

— T_4 is not a full m-ary tree for any m (as some have 2 children & some have 3 children)

Example

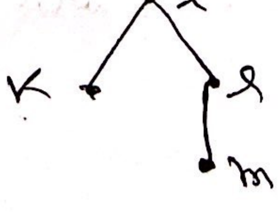


— The left child of d is f & right child of d is g

— left subtree of c

is

— right subtree of c is



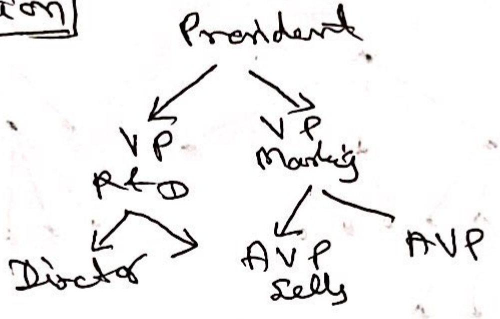
Trees as Models

① Saturated Hydrocarbons & Trees

Molecules

1857 Cayley discovered trees trying to enumerate the isomers of compounds of the form C_nH_{2n+2} (sat. hydrocarbon)

② Representing organization



③ Computer file systems

Properties of Trees

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

□ By Induction:

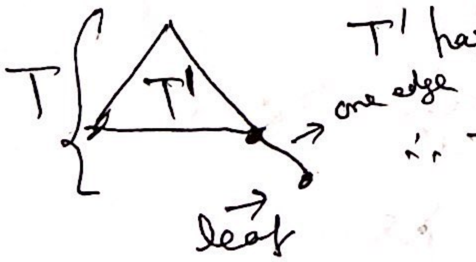
Base $n=1$ A tree with one vertex has 0 edges

$\therefore$ true for $n=1$.

I.H. ~~Suppose~~ Every tree with k vertices has $k-1$ edges ($k>0$ integer)

~~Suppose a tree T has k+1 vertices and~~
~~... is a leaf of T~~

Suppose T has $k+1$ vertices.



T' has k vertices so it will have $k-1$ edges

$\therefore$ T will have $(k-1)+1 = k$ edges

■

④ notes 11

Theorem: A full m-ary tree with i internal vertices contains $n = mi + 1$ vertices

□ Every vertex, except the root, is the child of an internal vertex. Now each of the i internal vertex has m children ⇒ there are mi vertices in the tree other than root. ∴ the tree contains $n = mi + 1$ vertices. ▨

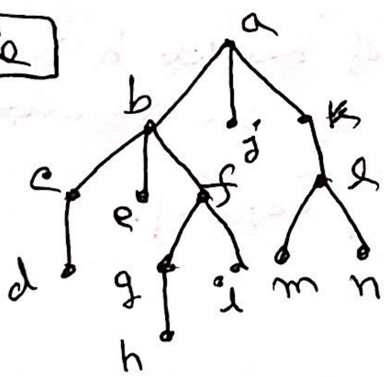
Theorem: A full m-ary tree with

(i) n vertices has $i = \frac{(n-1)}{m}$ 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.

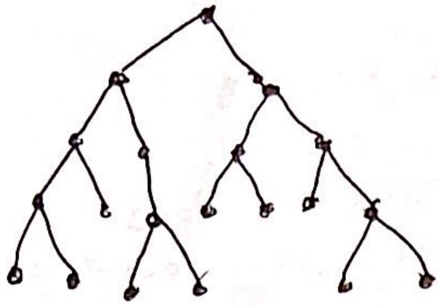
Example



- Root a is at level 0
- Vertices b, j, k are at level 1
- Vertices c, e, f & l are at level 2
- Vertices d, g, i, m & n are at level 3
- vertex h is at level 4
- ∴ the largest level of any vertex is 4 the tree has height 4.

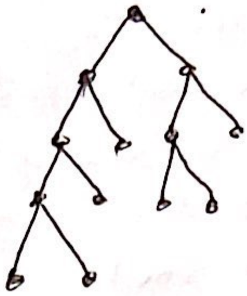
— A rooted tree of height h is balanced if all leaves are at levels h or $h-1$.

Example



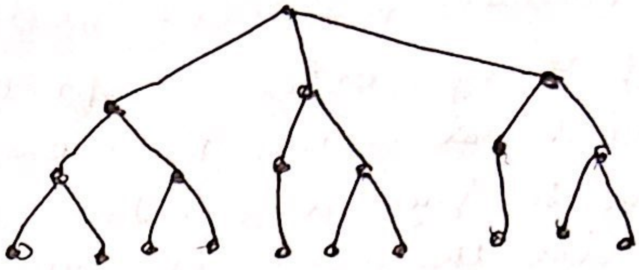
T_1

T_1 is balanced (∵ all its leaves at level 3 or 4)



T_2

T_2 is not balanced (∵ leaves at levels 2, 3 & 4)



T_3

T_3 is balanced (∵ all its leaves are at level 3)

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

□ Proof by induction on the height. ▨

Corollary 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 $h = \lceil \log_m l \rceil$.