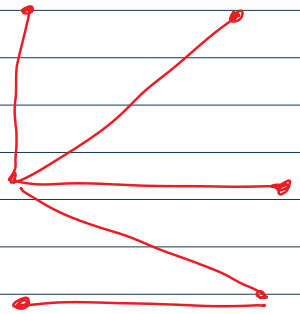
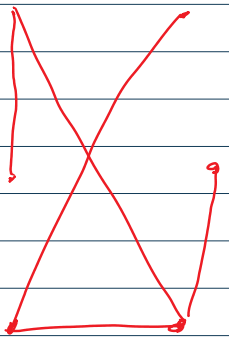
"Trees".

1- Connected
2- Undirected
3- No Simple Circuit.

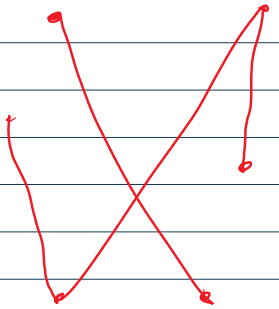
Ex1



(a) ✓

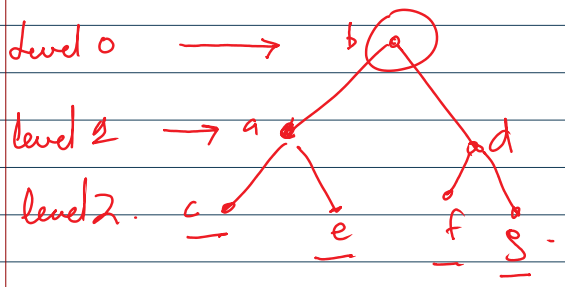


(b) ✓



(c) ✗

Rooted Tree:- 1) A Single Vertex designated as root.
2) Every Edge is directed away from Root.



A Simple Path exist btw every pair of Vertices which is Unique.

Parent → Unique, one level less.

Child → many, one level more.

Sibling → Same level, Same Parent.

Ancestor →.

Descendent →.

Right child →

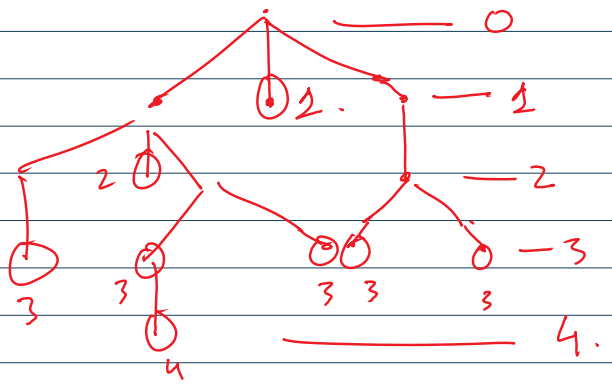
Left child → The at the left of the Parent.

Leaf. ⇒ No further child.

Level.

Height : Highest level of any Vertex.

Balanced. H, (H or H-1) all leaves.



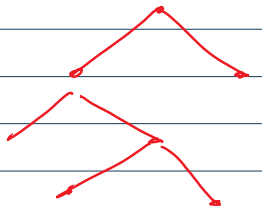
4.

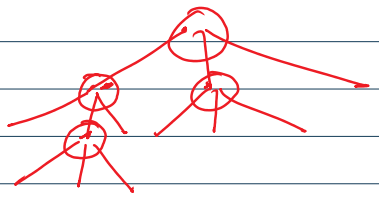
$H \geq 4$. $H-1=3$.

Ex 3 m-ary tree.

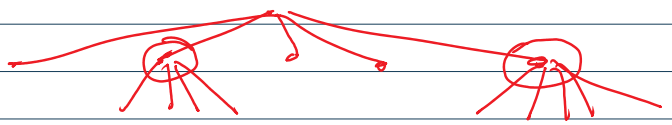
627.

2-ary tree.





3-ary tree.



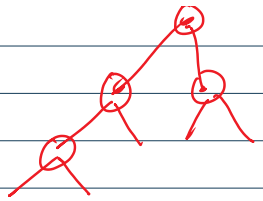
5-ary.

full m-ary tree.

full - 2-ary tree.

Tol.

124.



full - 2 - ary tree.
full - Binary tree.

theorem 2 : A tree with n vertices has n-1 edges.
630

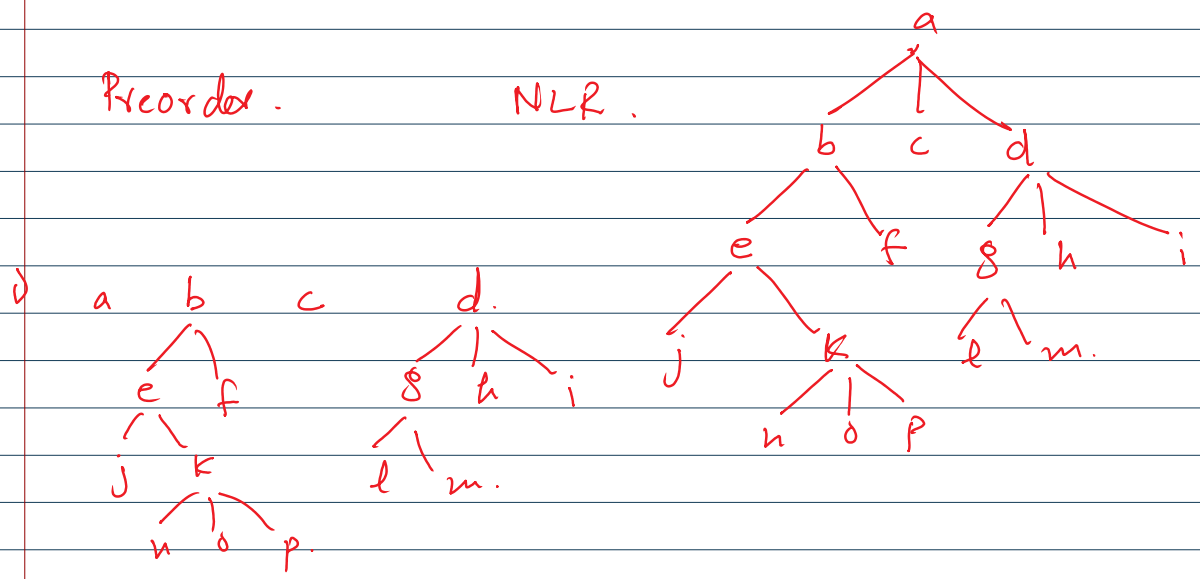
theorem 3 :- A full m-ary tree with i internal vertices contains $n = mi + 1$ vertices.
$= 2 \times 4 + 1 = 9$.

Ex 633
(1 - 40).

Applications. later on.

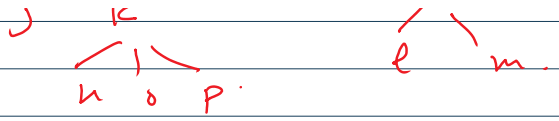
Tree traversal.

Preorder. NLR.

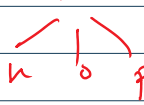


1) a b c d.
e f
g h i
j k
l m.
n o p.

2) a b e f c d g h i
j k
l m.
n o p.

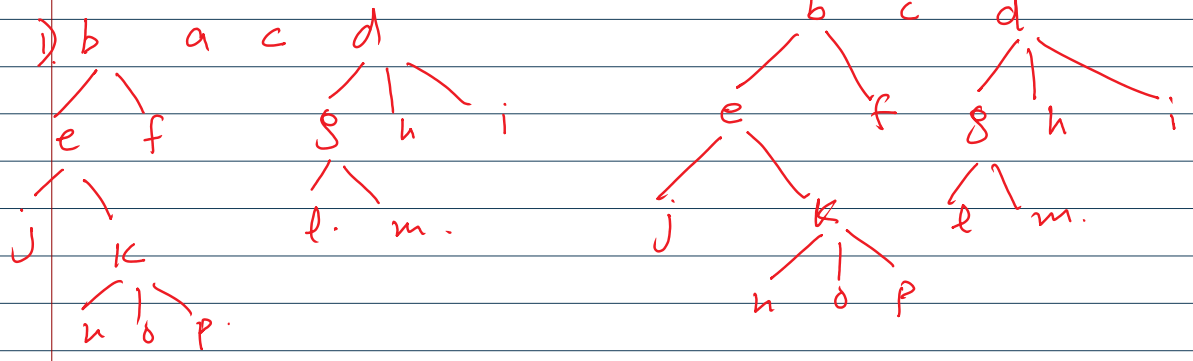


3) a b e j K f c d g l m h i



4 a b e j K n o p. f c d g l m h i

Inorder. LNR.
LN(L-R)..



2) e b f a c g d h i

3) j e K b f a c l g m d h i.

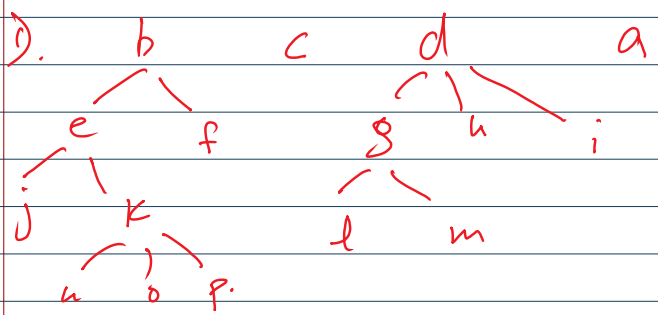
4) j e n K o p b f a c l g m d h i.

Pre order. NLR.
Inorder LNR.
Post order. LRN.

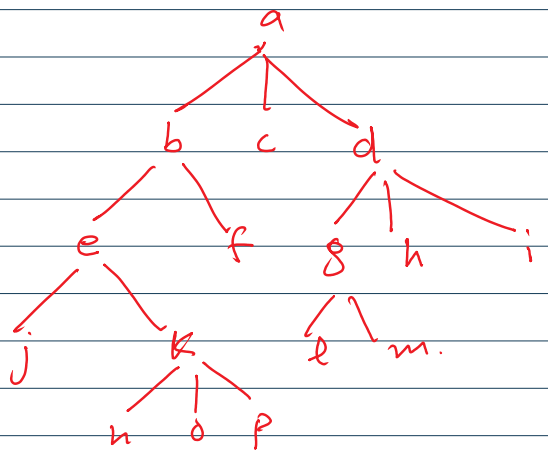
a

Inorder LNR.
Post order. LRN.

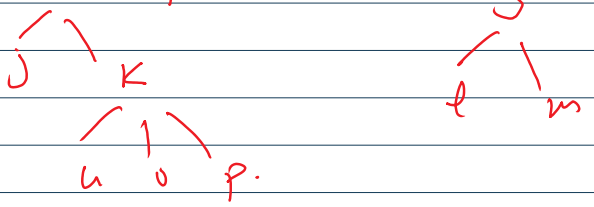
1).



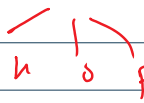
∴



2). e f b c g h i d a.



3) j k e f b c d m g h i d a.

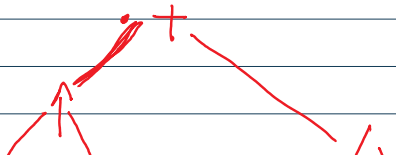


4) j n o p e f b c d m g h i d a.

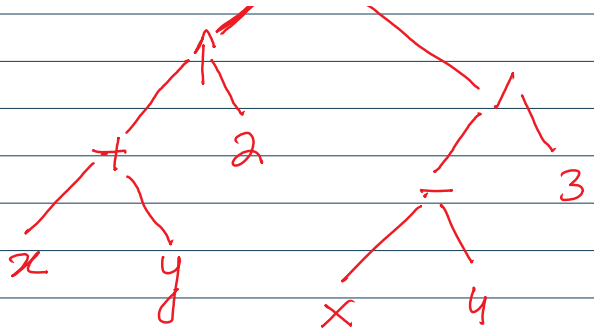
Tree traversal Application.

Evaluating Expressions.
- operators.
- operands.

Ex: $((x+y) \uparrow 2) + ((x-4)/3)$.



Preorder = Polish II. Prefix form
Inorder = Infix II.
Post order = Post fix



Inorder = Infix ||.
post order = Post from Reverse Polish

Ex7 :- P658

+ − * 2 3 5 / ↑ 2 3 4.

Prefix = Preorder = Polish.

+ − * 2 3 5 / 1 8 4

+ − * 2 3 5 2

+ − 6 5 2.

+ 1 2

3.

Ex8 P659

7 2 3 * − 4 ↑ 9 3 / +.

Postfix = Postorder. = Reverse Polish.

7 6 − 4 ↑ 9 3 / +.

1 4 ↑ 9 3 / +.

1 9 3 / +.

1 3 +

4.

(P660-662 1-45) Ex:-