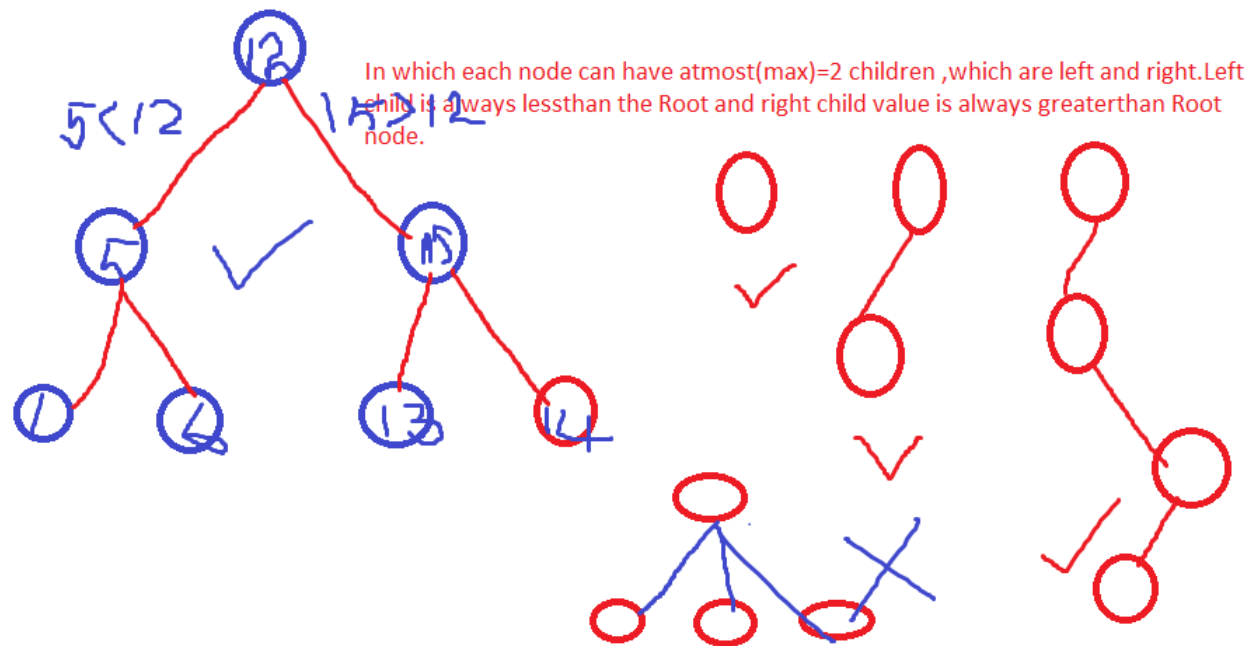
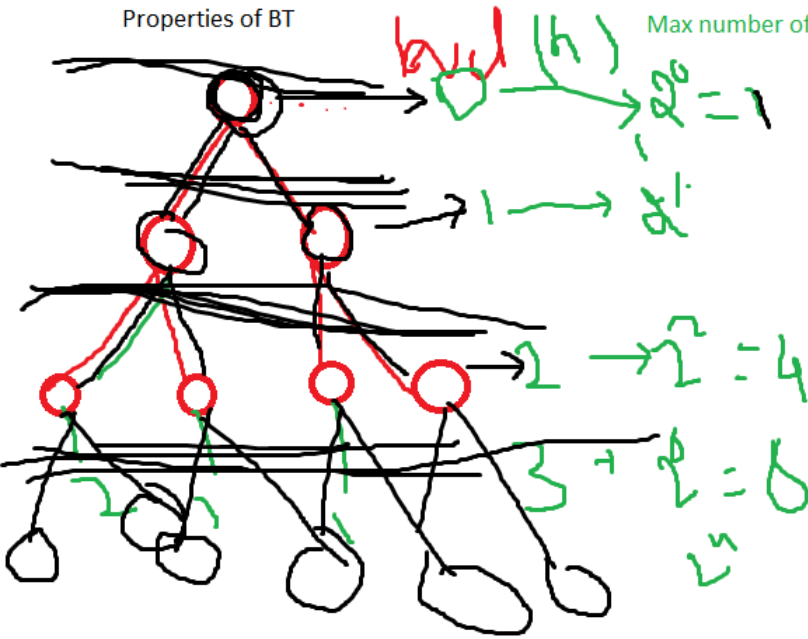


Binary Tree & Types & Properties



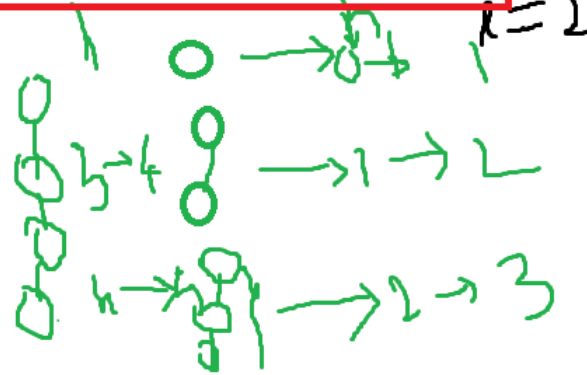
Properties of BT

Max number of nodes possible at any level 'h' is 2^h



$2^0 + 2^1 + 2^2 + \dots + 2^h = \text{Geometric Progression}$
 $2^{h+1} - 1$

Min Number of nodes of a height (h) $\rightarrow h+1$



$$h) \quad 2^0 + 2^1 + 2^2 + \dots + 2^h = G.P$$

$$2^{h+1} - 1 \quad \text{or} \quad 2^h - 1$$



$h+1$



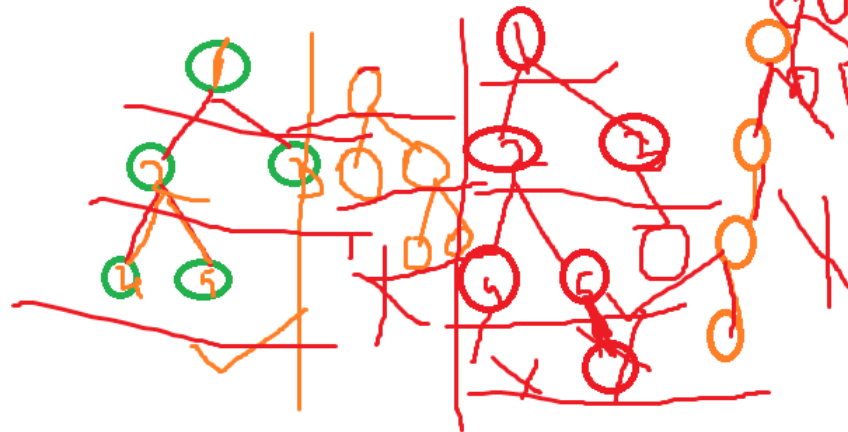
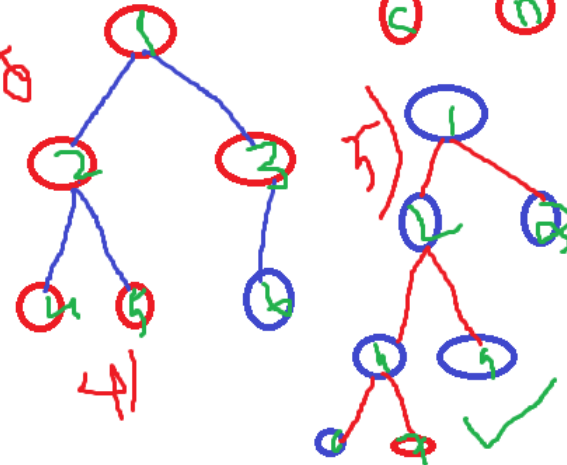
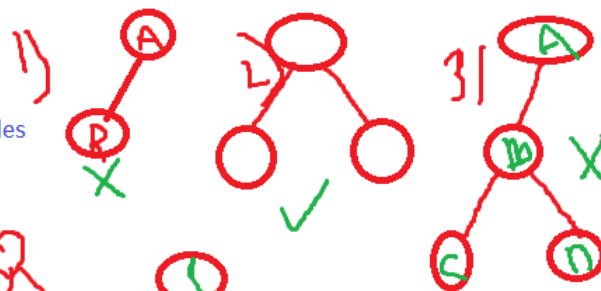
Types of Binary tree

Full/Proper/Strict BT If each node have either zero or 2 children nodes

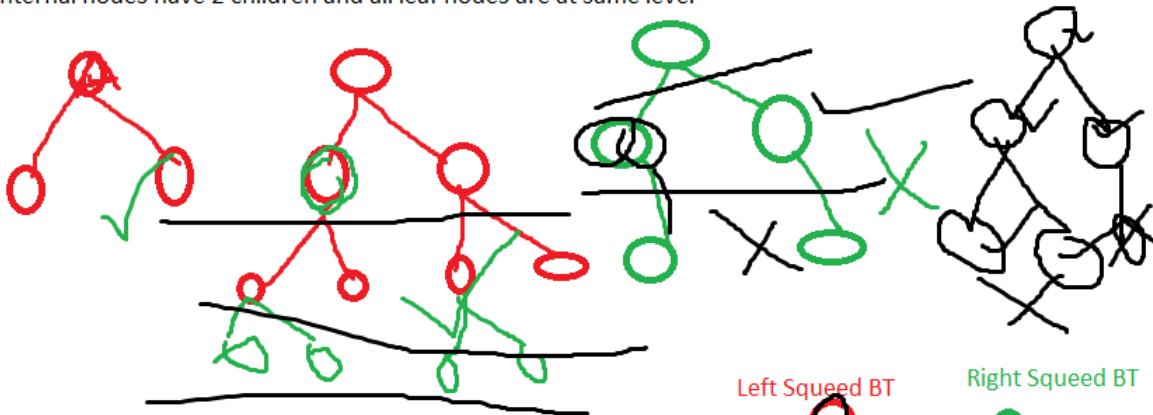
Complete Binary tree If All levels are completely filled

Perfect BT

Degenerate(Pathological) BT



3) Perfect=All Internal nodes have 2 children and all leaf nodes are at same level



4) Degenrate(Pathological) BT---->Each Internal Node is having one child

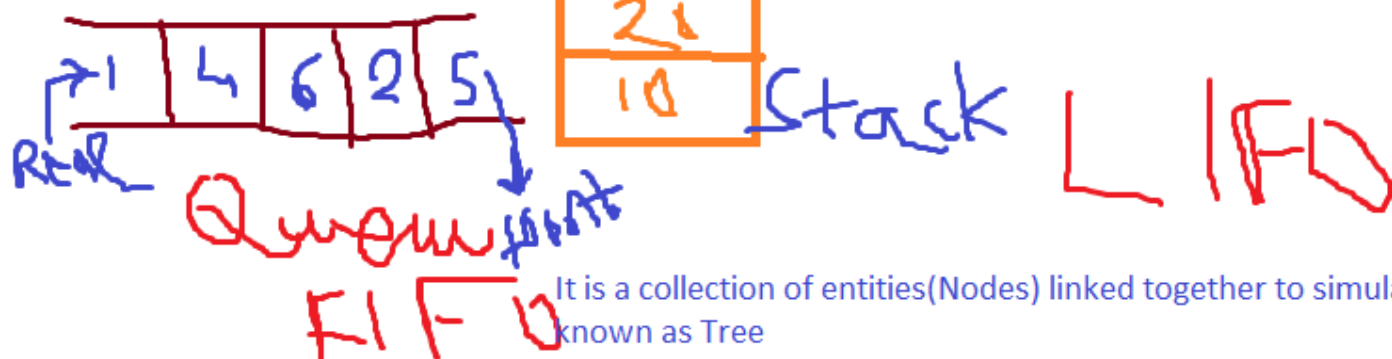
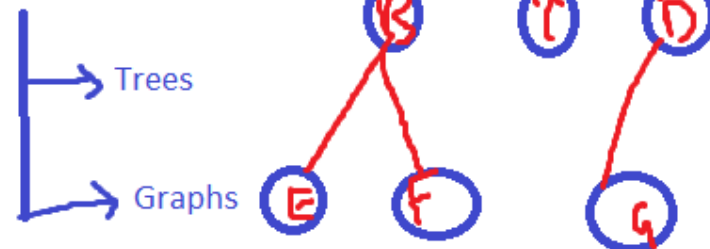


Data Structures

Linear data Structures



Non-Linear Data Structures

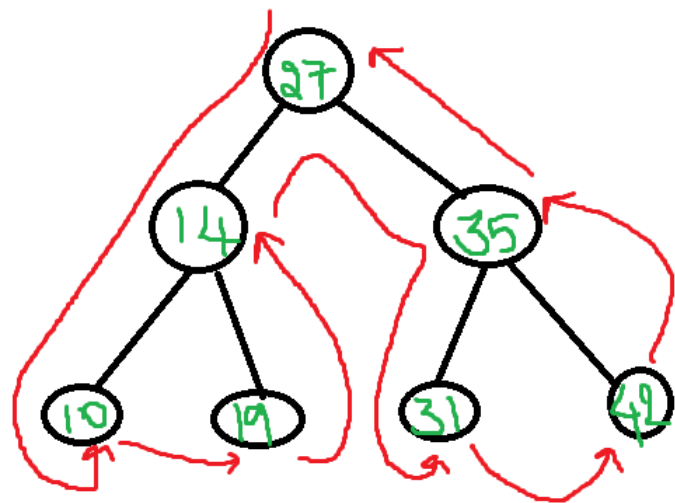


It is a collection of entities (Nodes) linked together to simulate a Hierarchical Structure is known as Tree

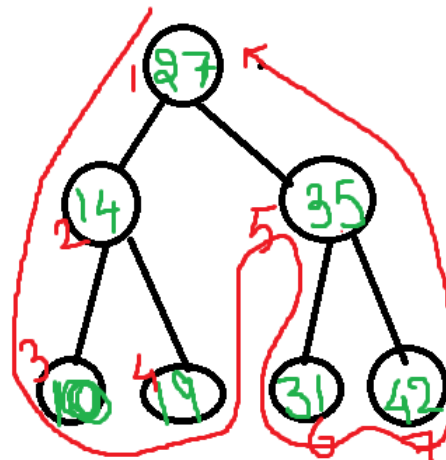
Post Order Left--->Right--->Root

PreOrder Root--->Left--->Right

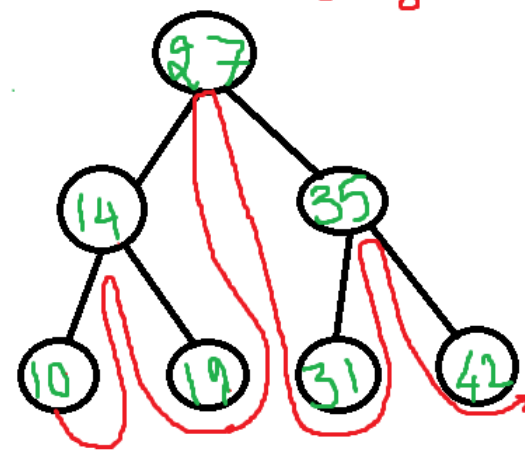
InOrder Left--->Root--->Right



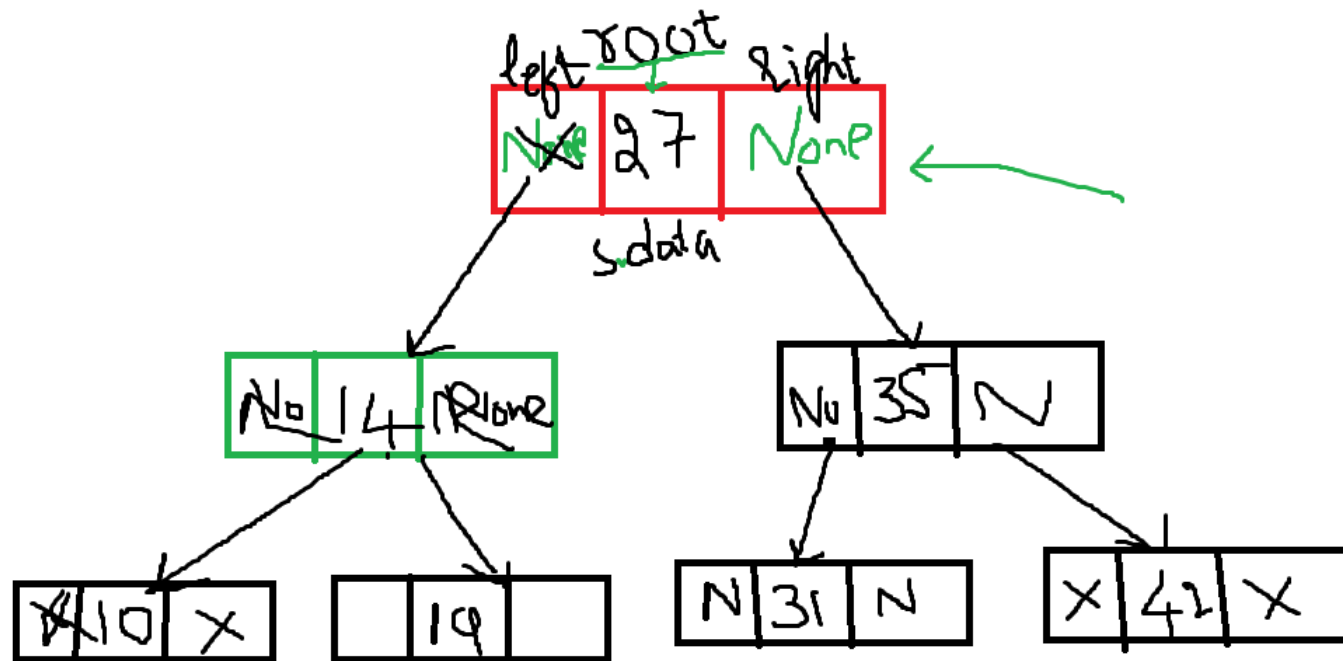
[10,19,14,31,42,35,27]



[27,14,10,19,35,31,42]



[10,14,19,27,31,35,42]



Properties of Trees

Root----->There is no parent node

parent----->Immediate predecessor is

Child--->Immediate Successor is a child

Nodes--->Nodes are A,B,C,D,E,F,G,H,I,J

Leaf(External Nodes)--->The Node which is not having any child[C,f,i,j,g,h]

Non-Leaf(Internal Nodes)--->The node which is having atleast one child[A,B,D,E,F]

Path--->Sequence of Consecutive edges from Source to Destination



Ancestor---> Any predecessor node on the path from root to that node-->Ancestor(H)=A& D and Ancestor(J)=A,B,&E

Sibling-->All the Children of Same parent is a Sibling-->Siblings(A)--->B,C,D

Decendant-->Any Successor node on the path from that node to leaf node-->Decendant(e)=i,j Decendant(B)=E,F and I,J

Degree-->The number of childrens of that node is a degree of a node

Degree of a leaf node is always Zero(0)

Degree of a tree is max degree among all nodes

Depth of a node-->Length of path from root to that node --->Depth(E)= 2, D(J)=5

Height of a Node-->Number of edges in the longest path from that node to leaf-->Height(B)= 2, H(A)=3

