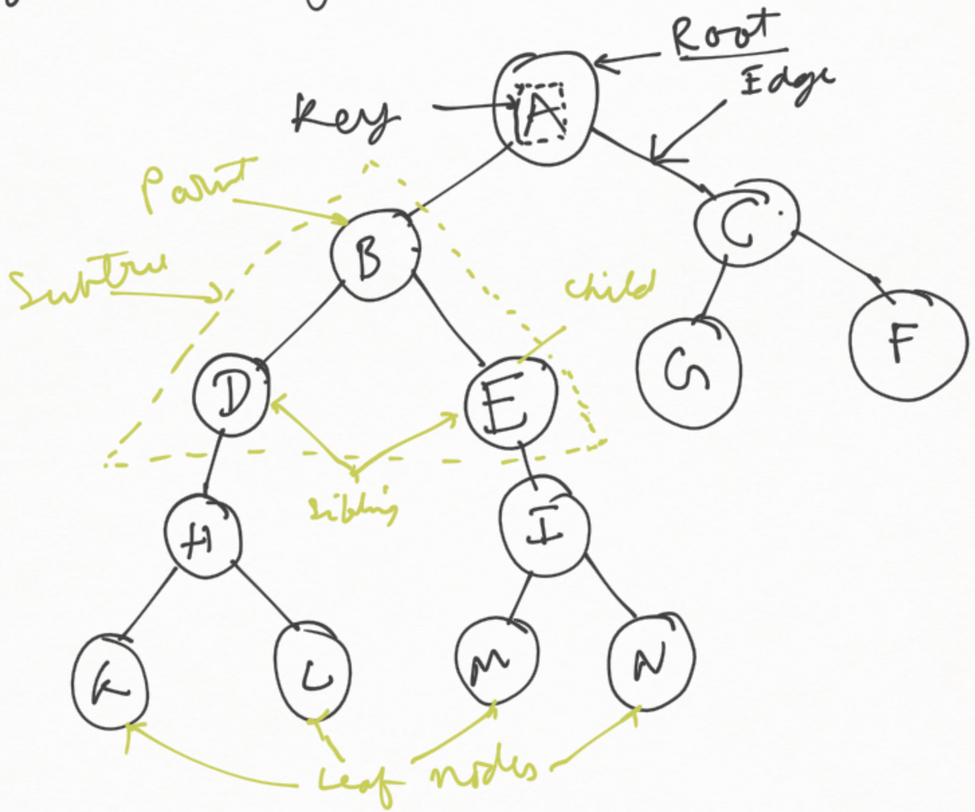
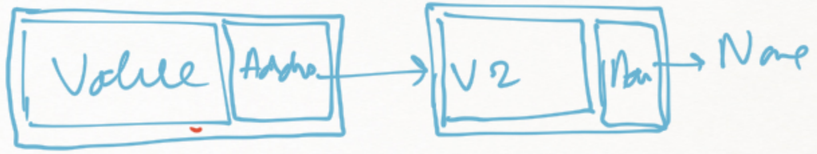
Tree Data Structure

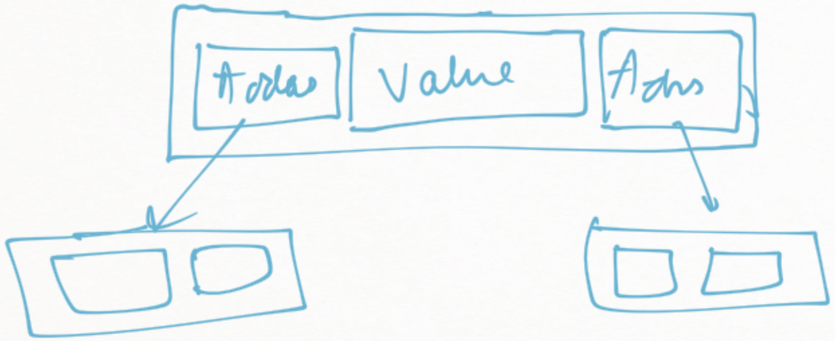
A tree is a non linear hierarchical data structure which consists of nodes connected by edges

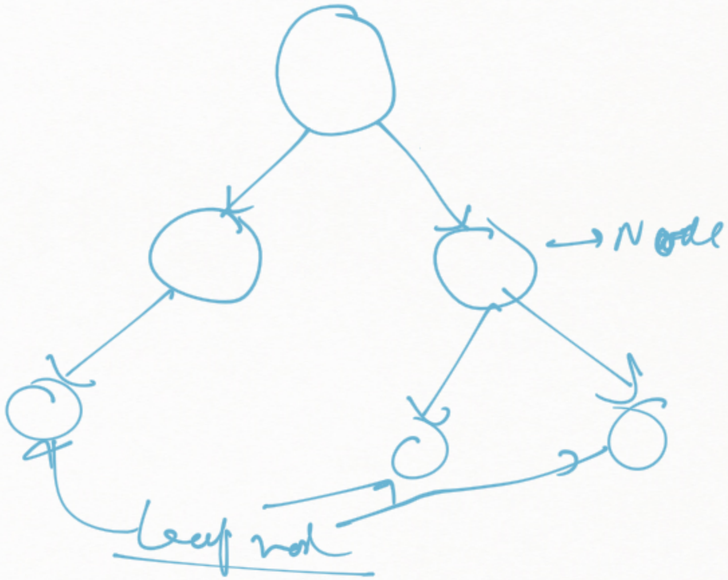


leaves

Node: It contains a key or value and pointers to child node.





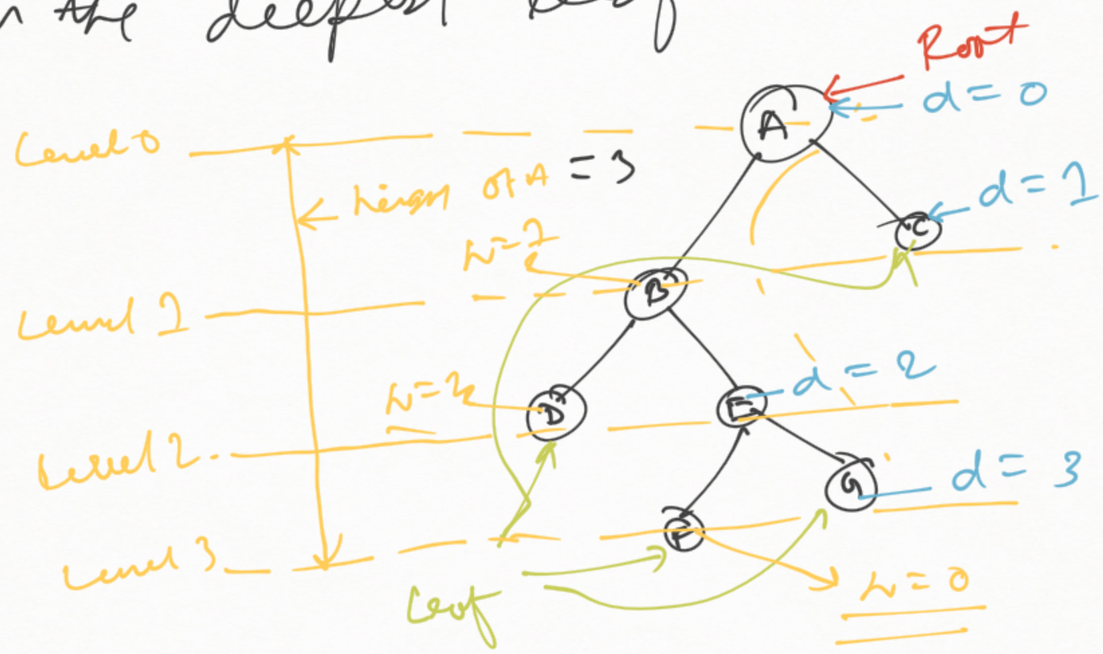


Edge : It is the link b/w any two Nodes

Root : It is the topmost node of a tree.

Height of a Node : The height of a Node is the number of edges from the deepest leaf

Depth of a Node : the number of edges from root to Node

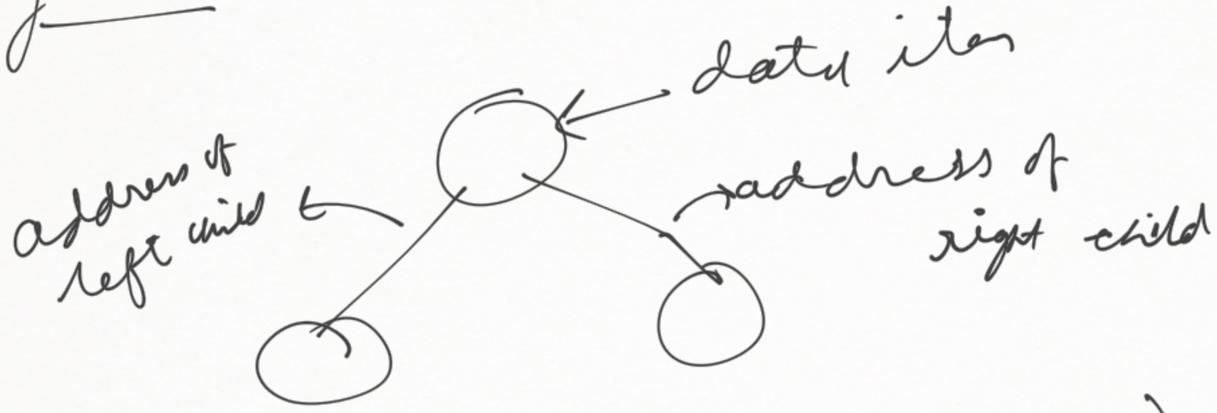


leaf nodes :
D, F, G, C

Degree of a Node: The total number of branches of the node

Forest: A collection of disjoint tree is called as forest.

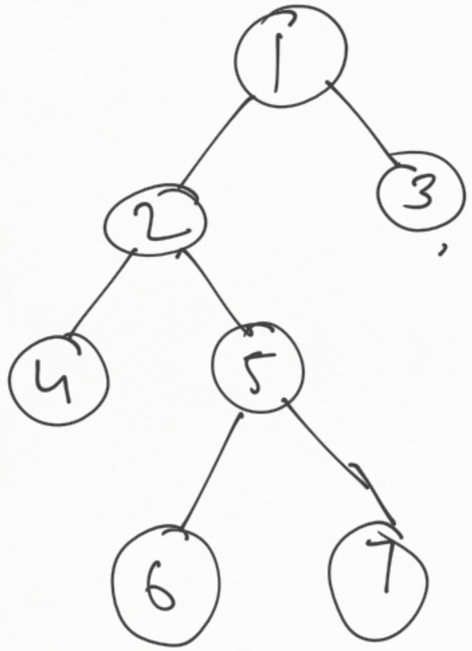
Binary tree



1) at most two children
2) Each node will have
   data item
3) address of left child
4) address of right child.

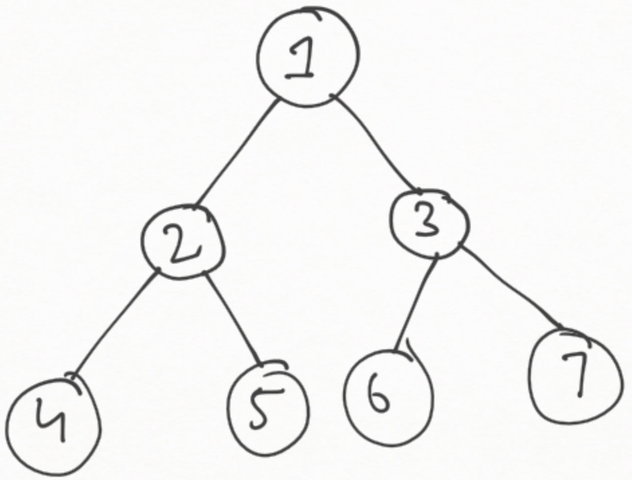
Types of binary tree

1. Full Binary tree



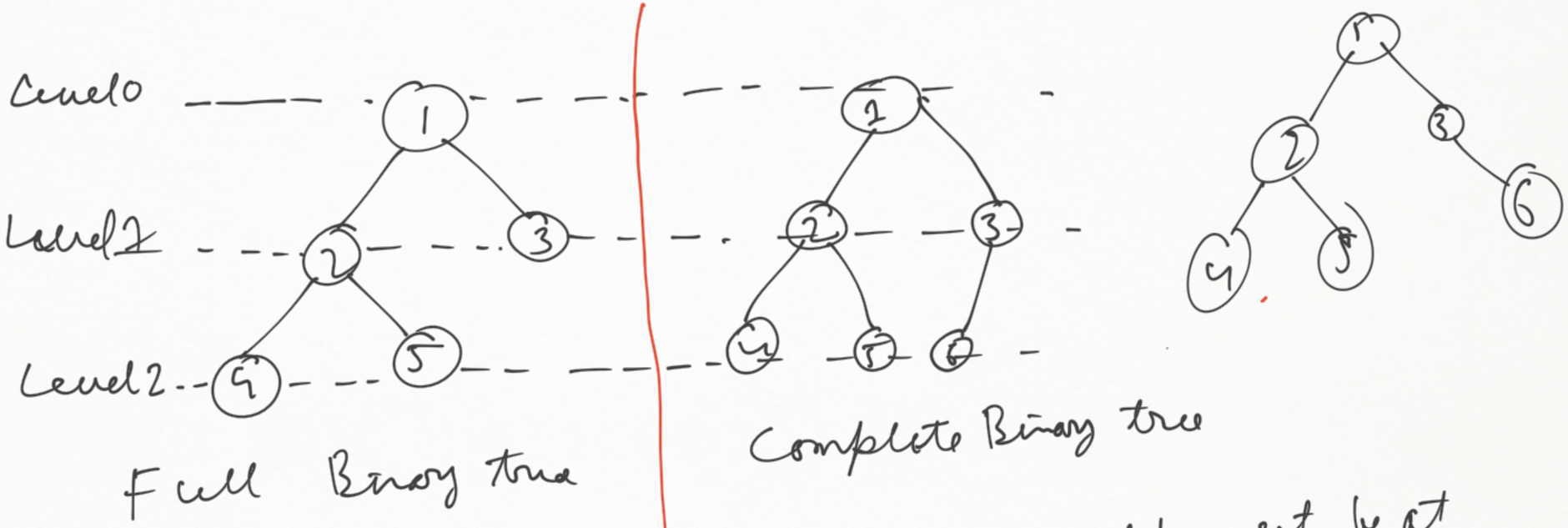
1) every parent have 0 or 2 child nodes

2. Perfect Binary tree:



1) every internal node has exactly 2 child nodes
2) all the leaf nodes are at same level

3) Complete Binary Tree:



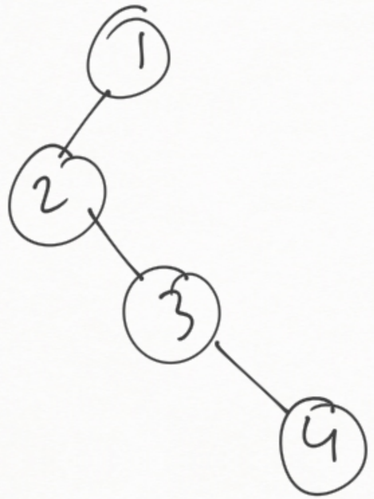
Full Binary tree

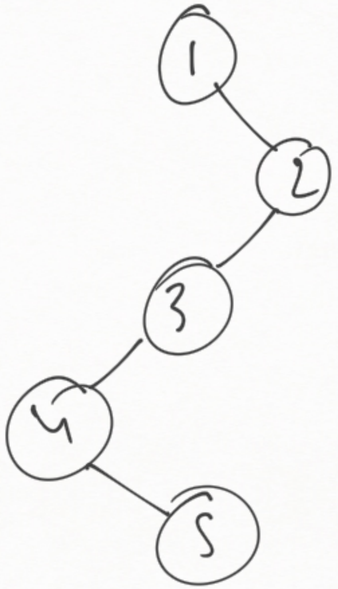
Complete Binary tree

1) Every child must be at same level

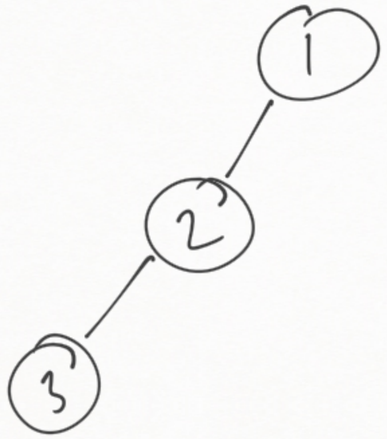
2) leaf element lean towards left

4. Degenerate or Pathological Tree

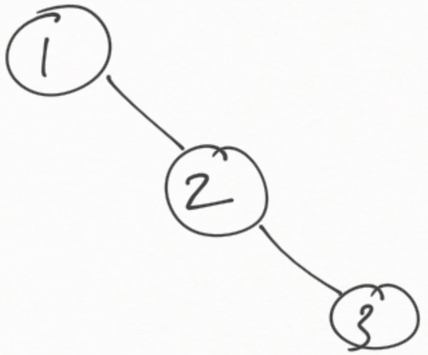




5) Skewed Binary Tree

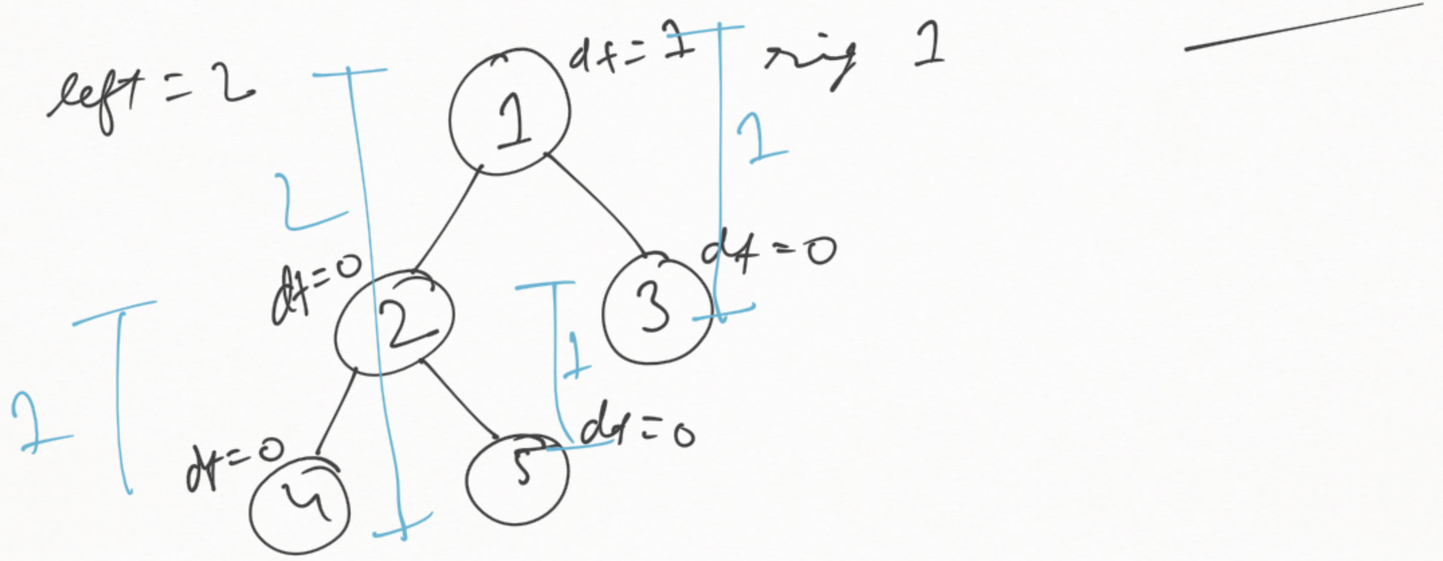


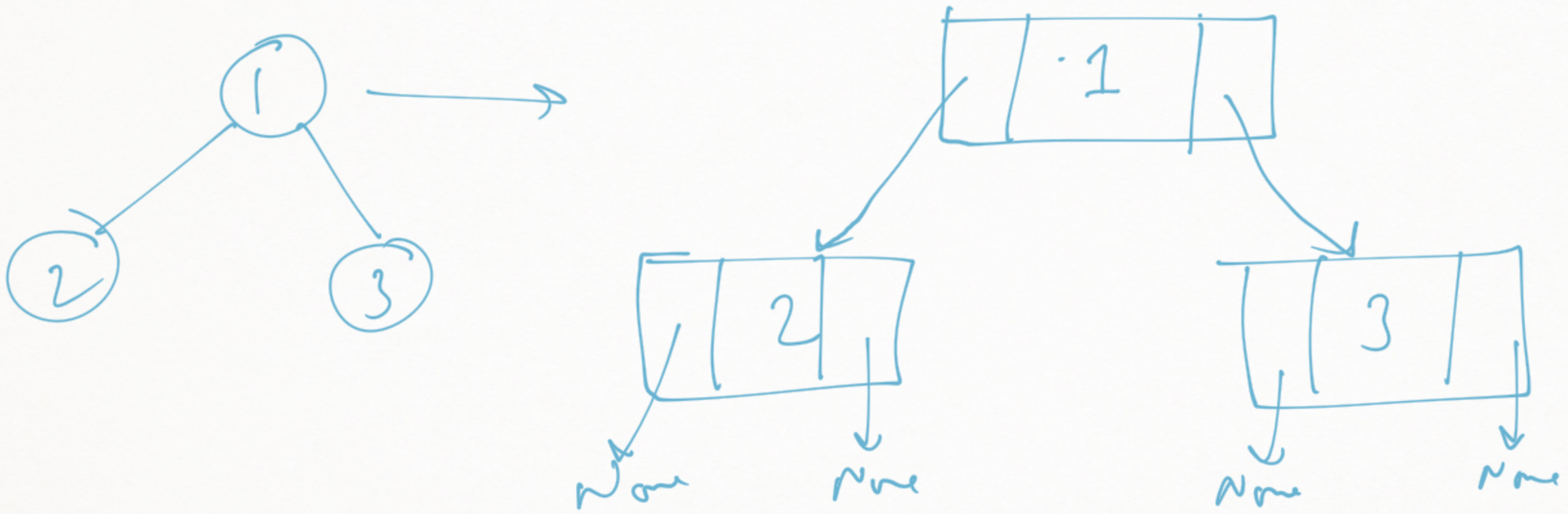
left-skewed binary tree



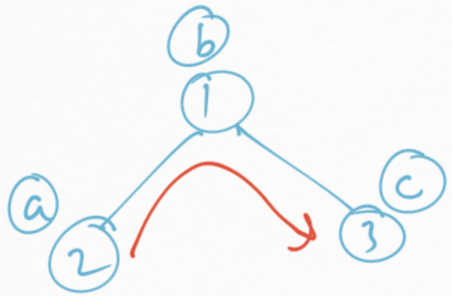
right-skewed binary tree.

6. Balanced Binary tree.




1
2
3
1
2
3
None
None
None
None

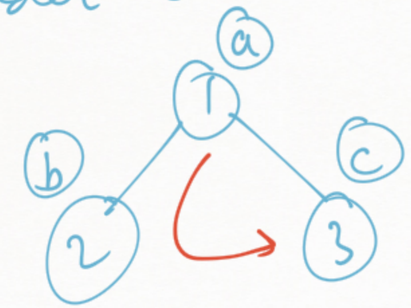
inorder traversal



a → b → c

2 → 1 → 3

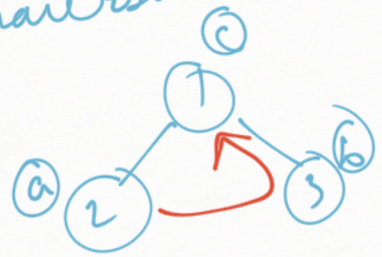
preorder traversal



a → b → c

1 → 2 → 3

postorder traversal



a → b → c

2 → 3 → 1

60%-70%

10 - 20%

20 - 30%

30 - 40%

40 - 50%

50 - 60%

:

90 - 100%