

Definition

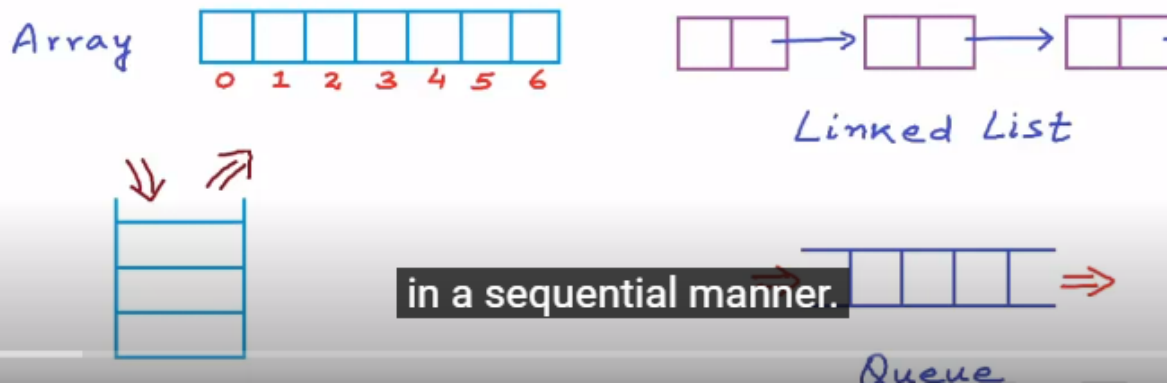
tree:

Graph G is called a tree if G is connected and contains no cycles.

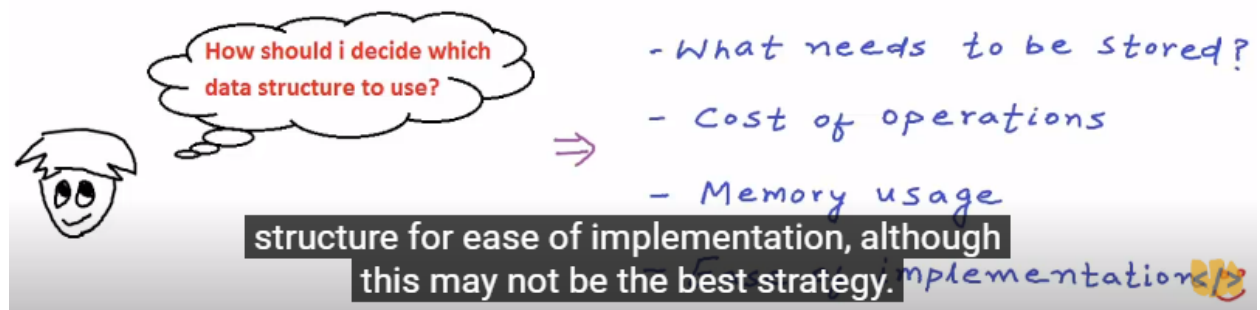
- ▶ Graph whose connected components are trees: forest

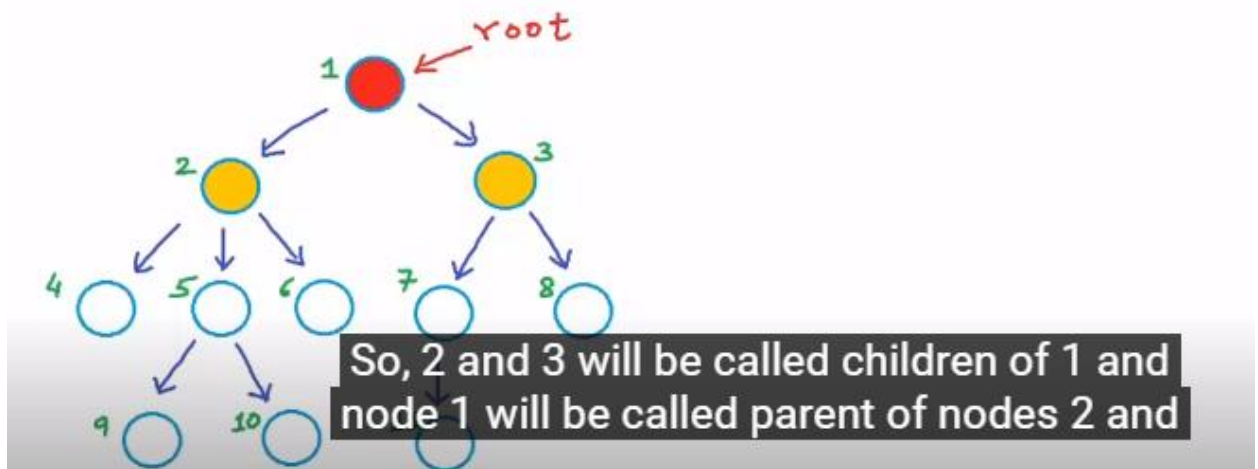
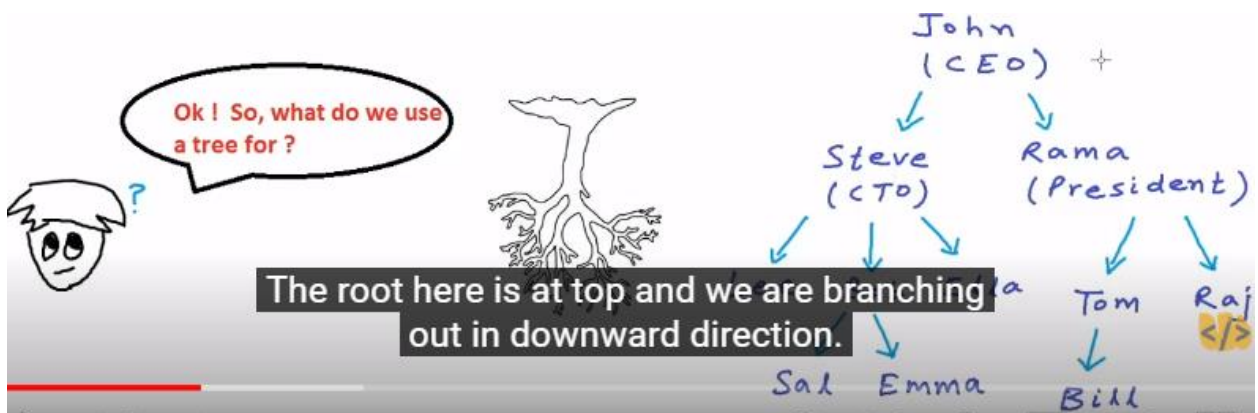
Introduction to Trees

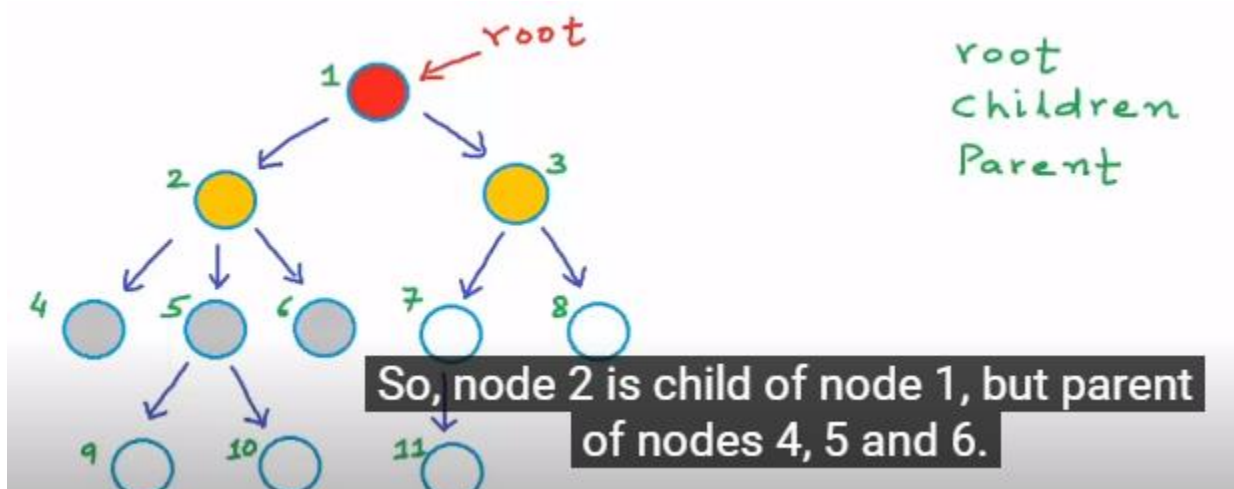
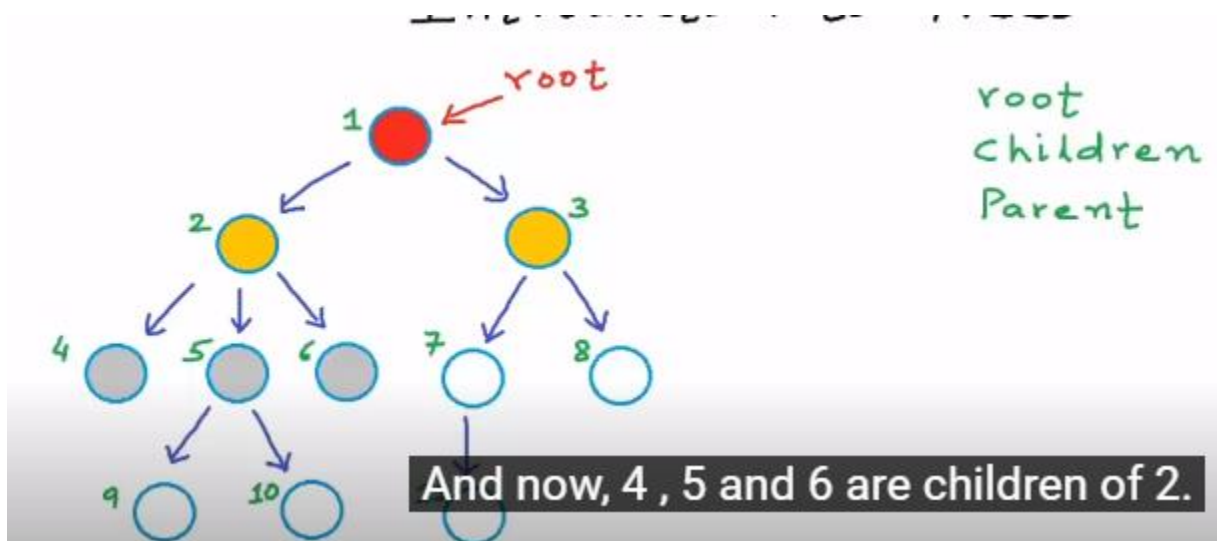
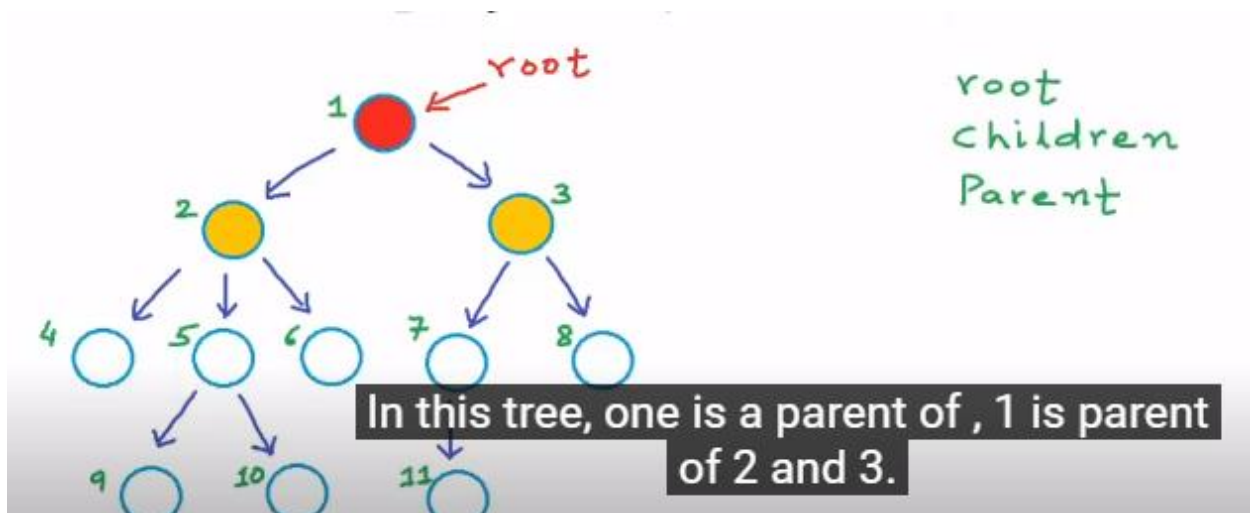
Linear data structures:

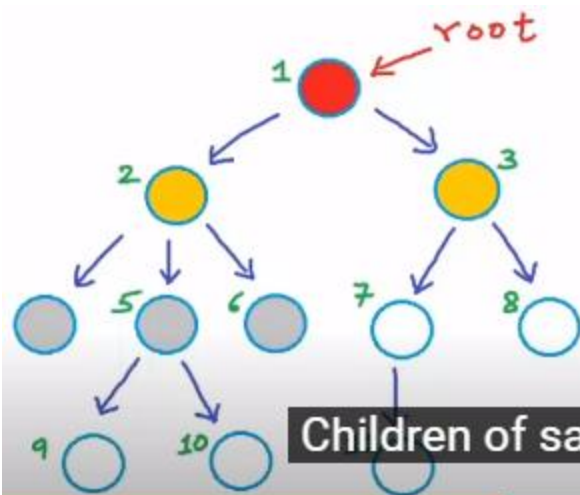


Introduction to Trees



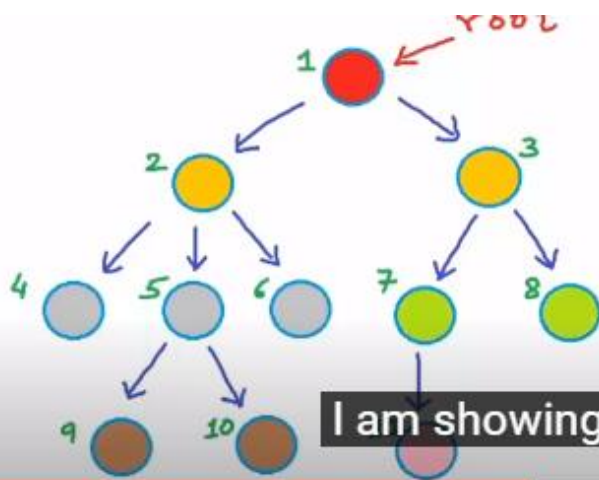






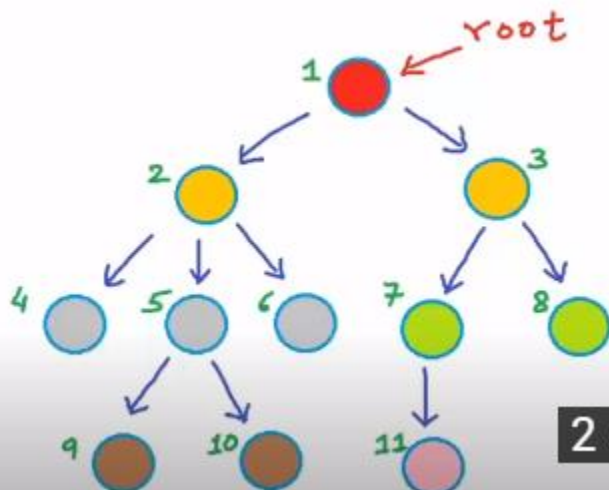
root
children
Parent
Sibling

Children of same parent are called sibling.



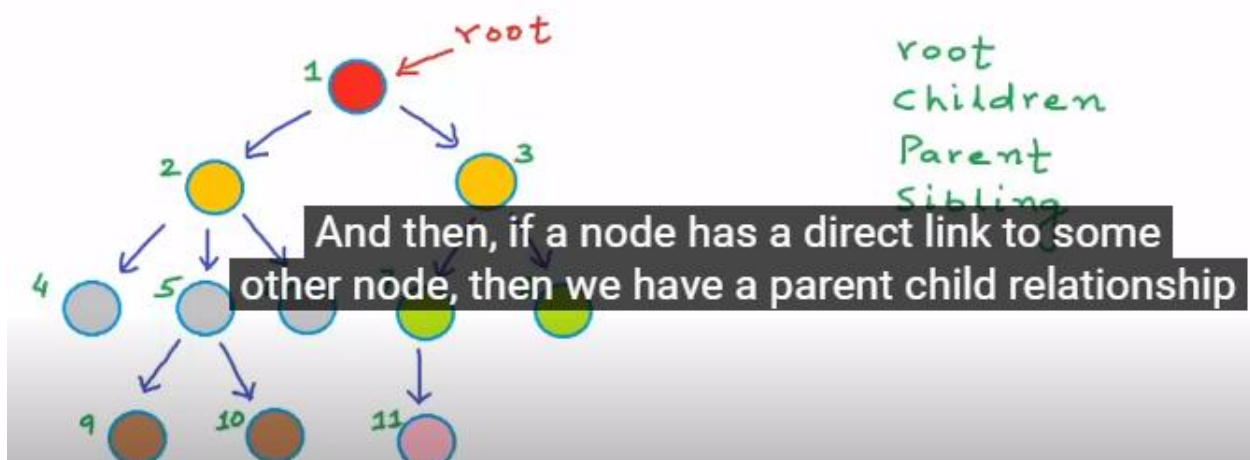
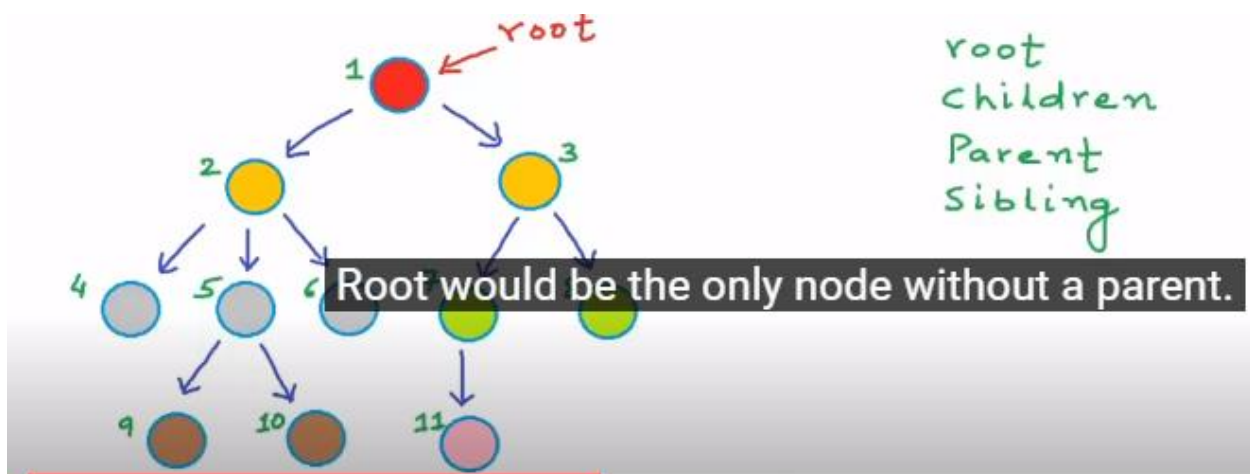
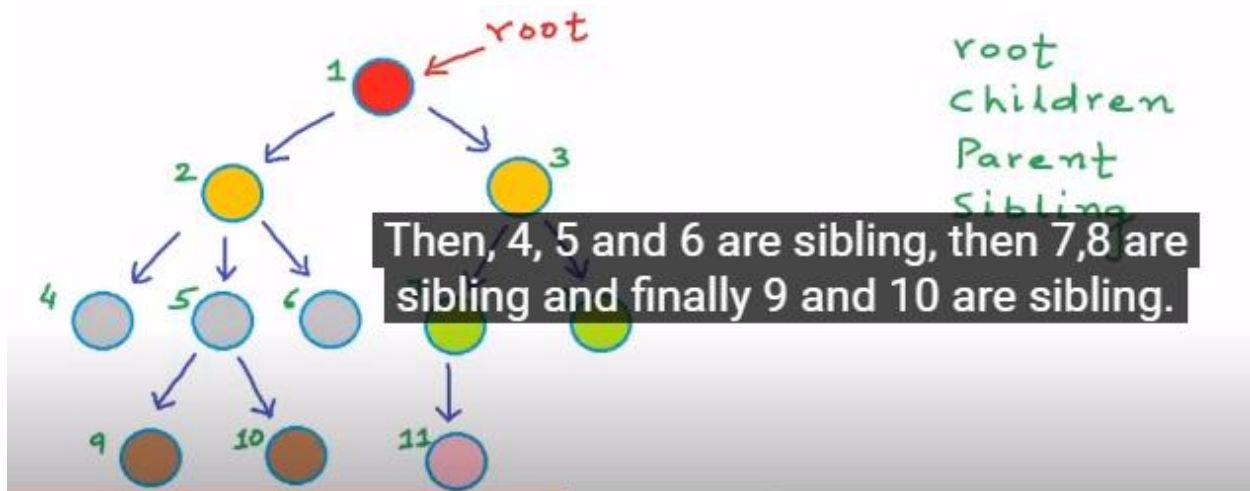
root
children
Parent
Sibling

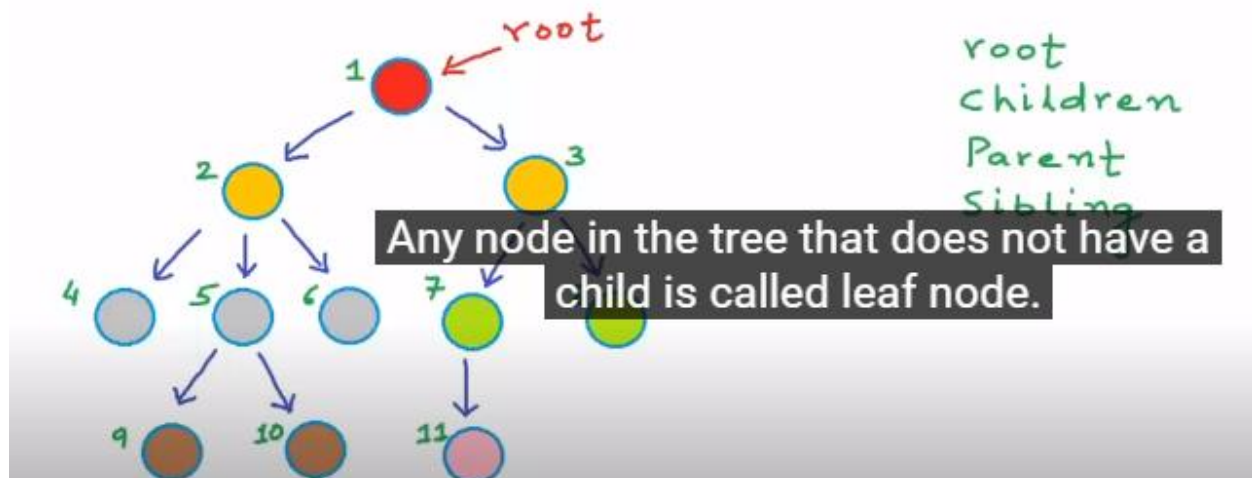
I am showing siblings in same color here.



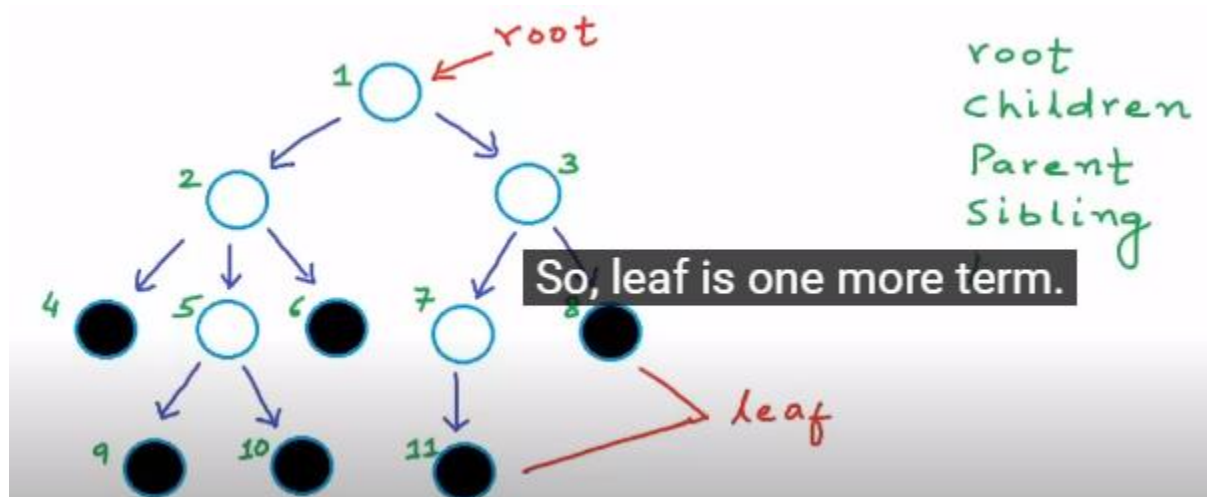
root
children
Parent
Sibling

2 and 3 are sibling.

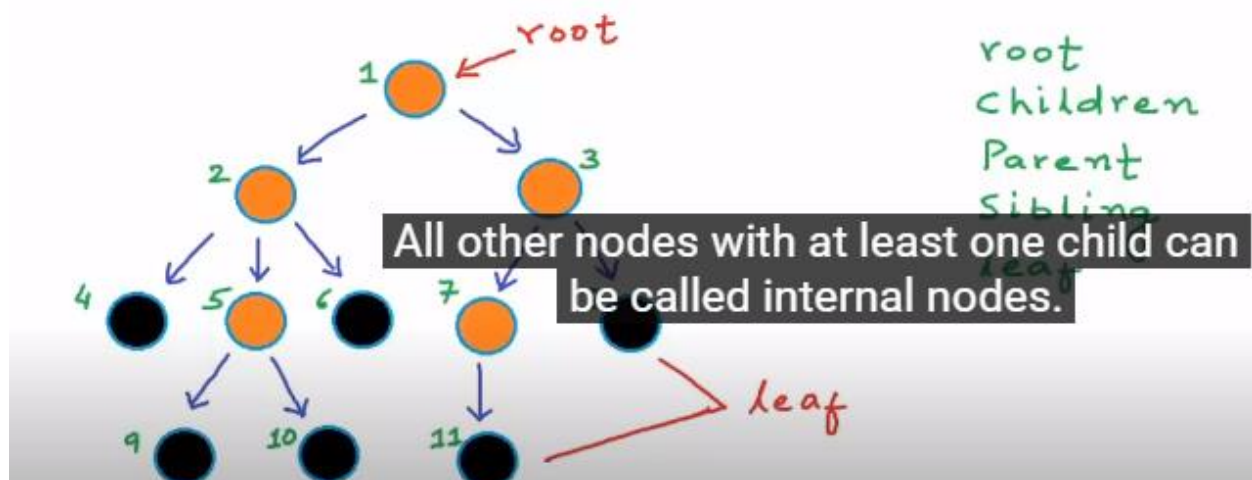




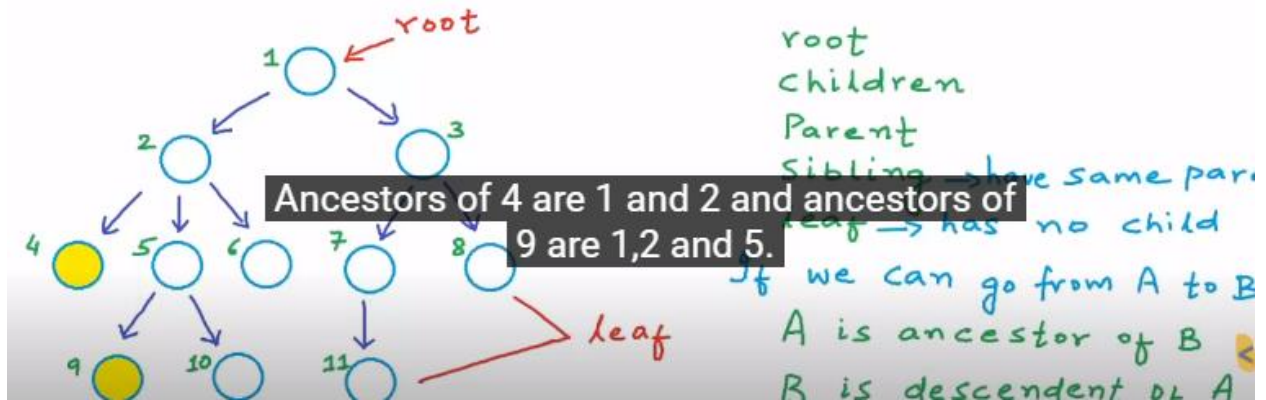
root
children
Parent
Sibling



root
children
Parent
Sibling

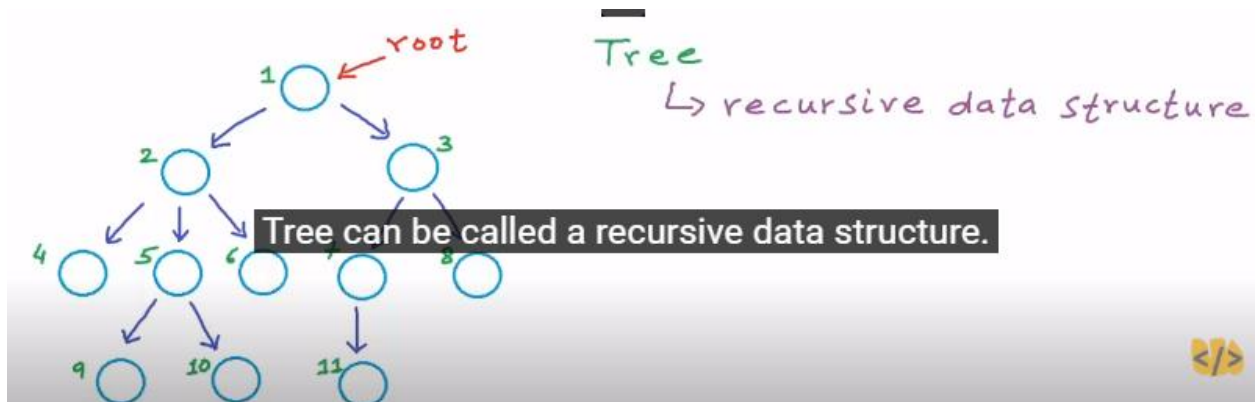


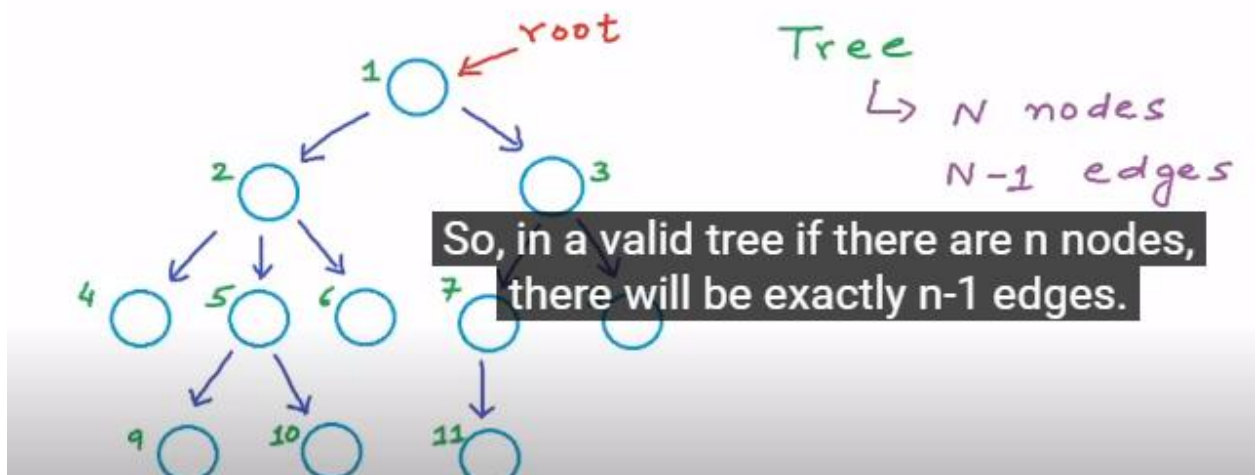
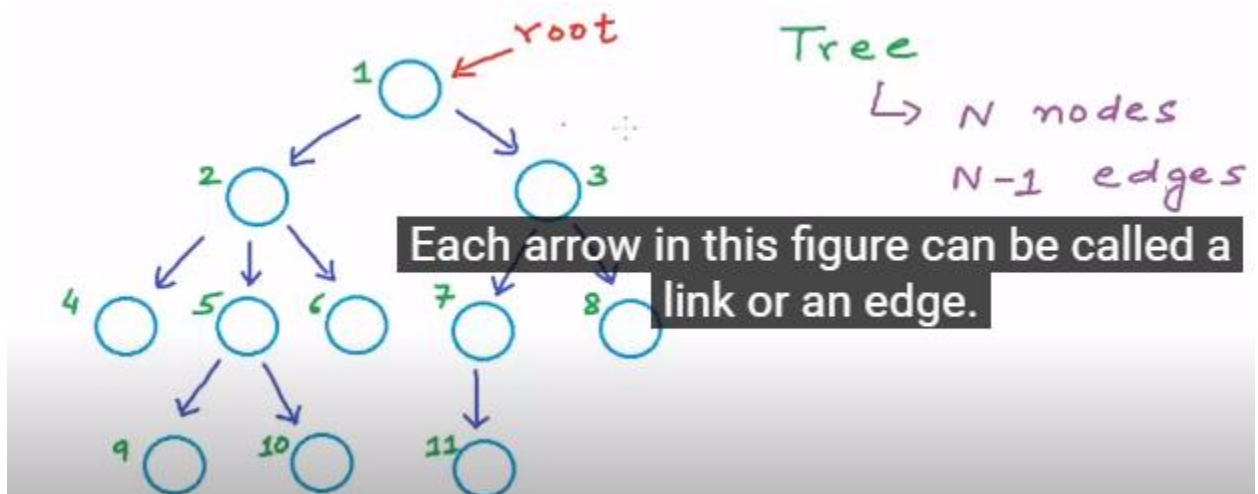
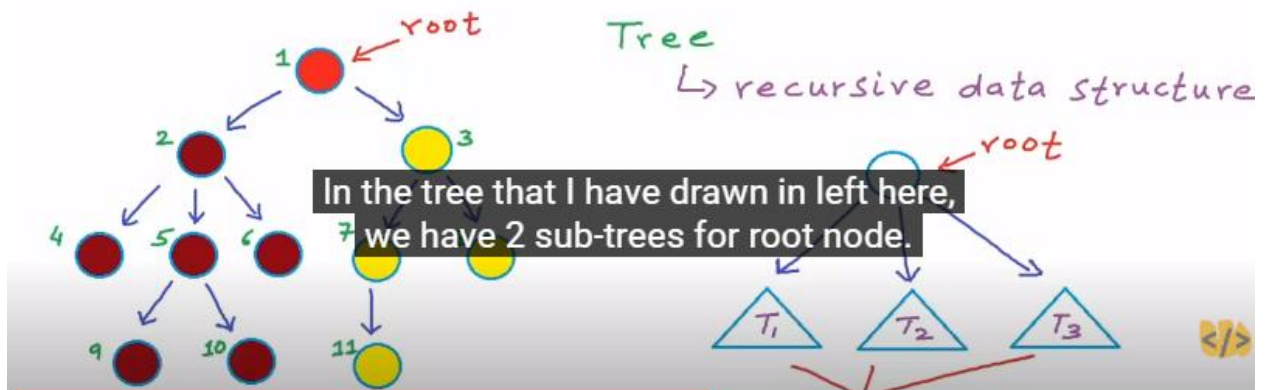
root
children
Parent
Sibling
leaf



root
children
Parent
Sibling
leaf
ancestor
descendent
cousin

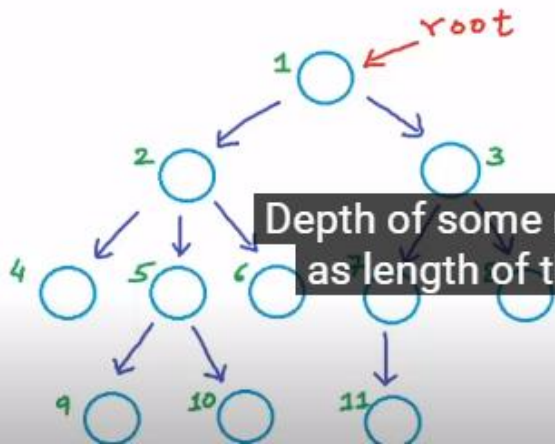
ome properties





Depth and Height

Depth of $x =$



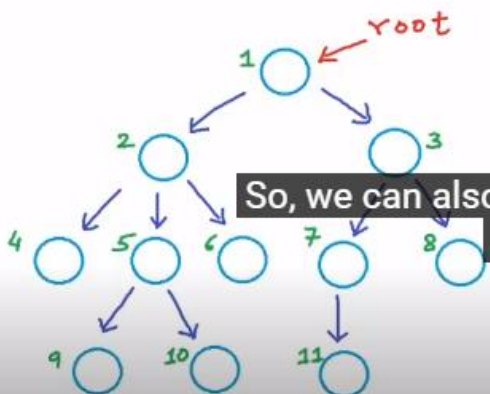
Depth of some node X in a tree can be defined as length of the path from root to Node X .

Depth and Height

Depth of $x =$

length of path from

So, we can also say number of edges in path from root to X .



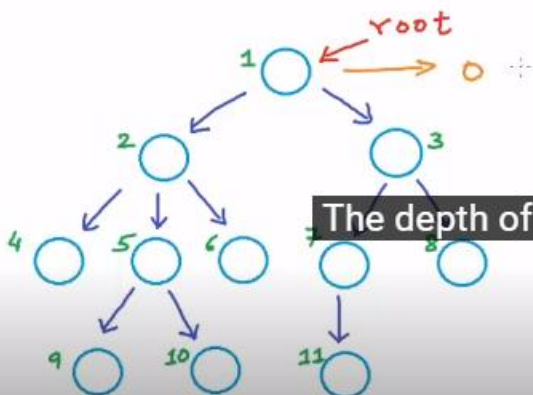
OR
No. of edges in path from root to x

Depth and Height

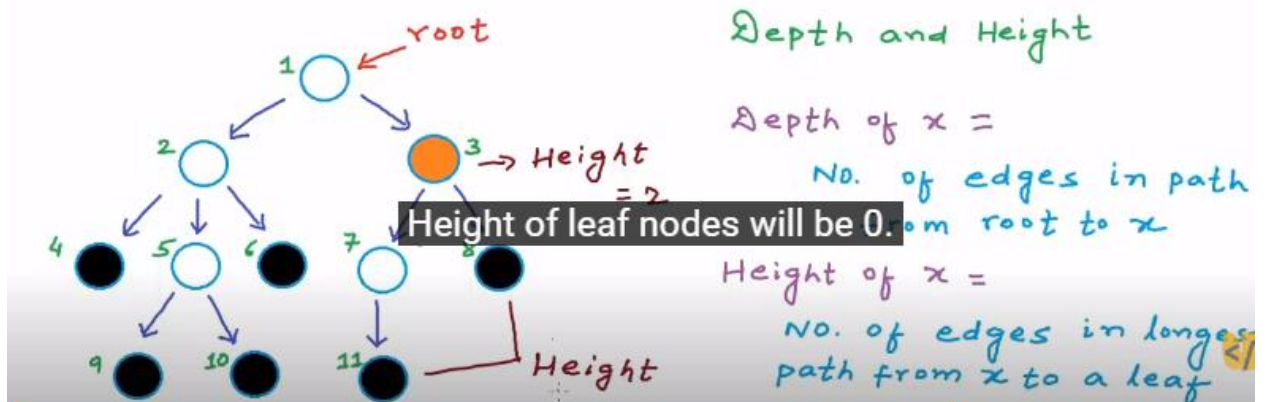
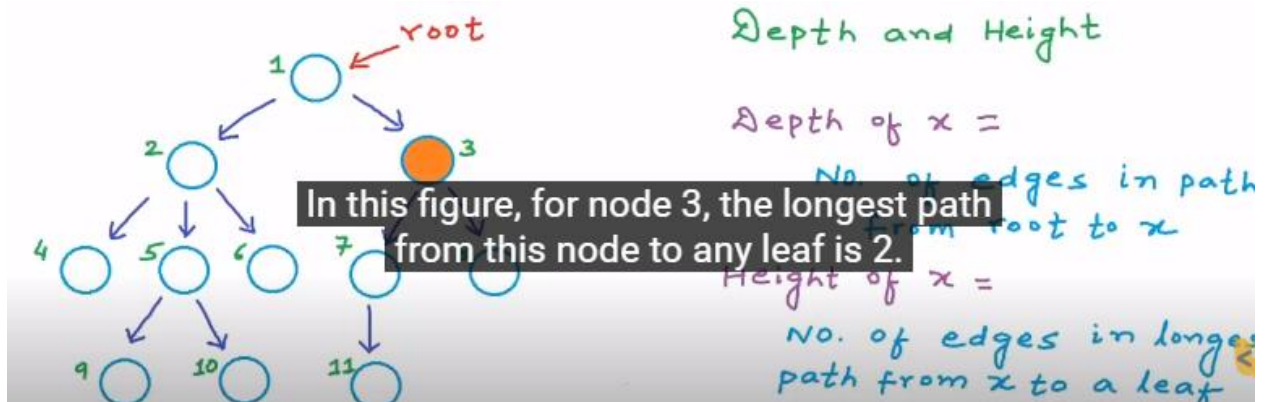
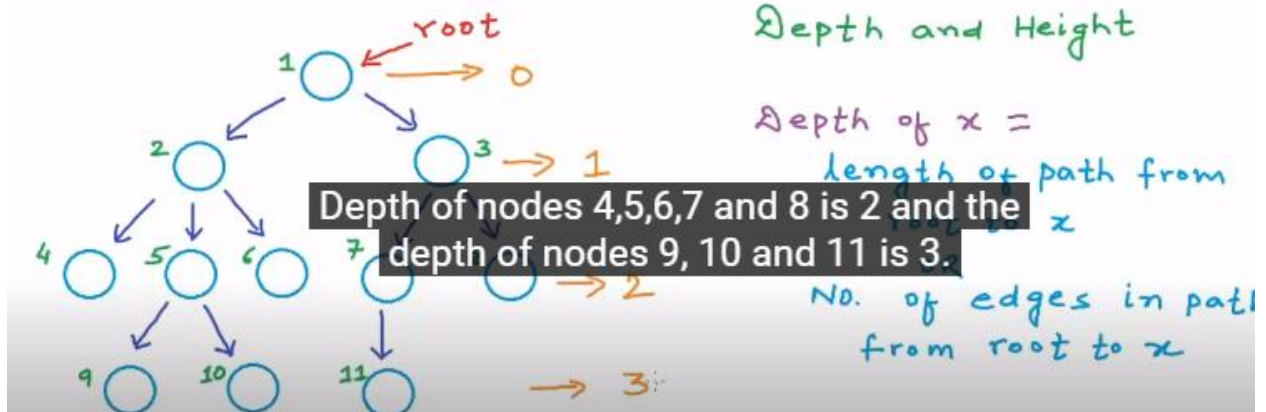
Depth of $x =$

length of path from to x

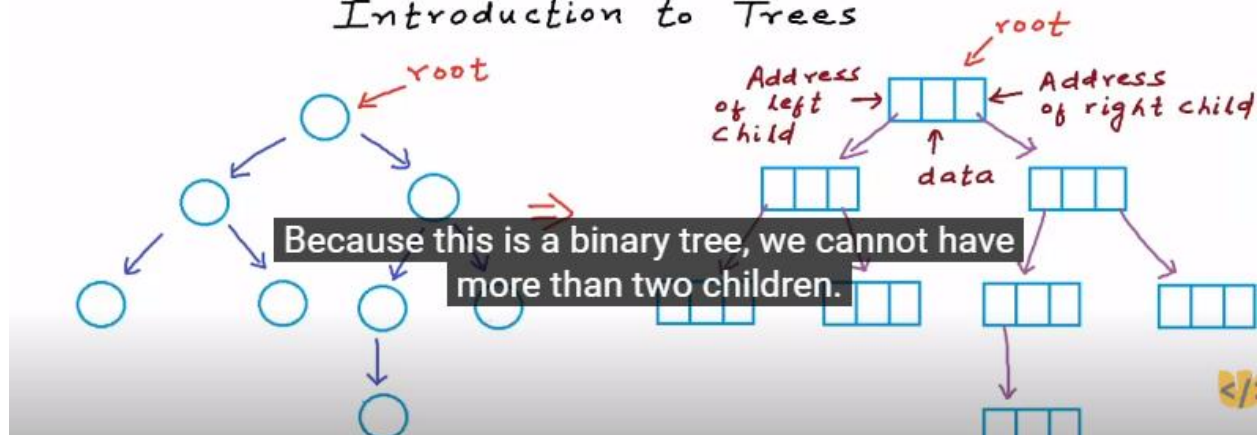
The depth of root node will be zero.



OR
No. of edges in path from root to x



Introduction to Trees



Applications:

- 1) Storing naturally hierarchical data → eg:- file system

For example, the file system on your disc drive, the file and folder hierarchy is naturally

Applications:

- 1) Storing naturally hierarchical data → eg:- file system

- 2) Organize data for quick search, insertion, deletion
Its really fast and efficient and is used for dynamic spell checking.

→ eg:- Binary Search trees

- 3) Trie → dictionary

Applications:

1) Storing naturally hierarchical data → eg:- file system

2) Organize data for quick search, insertion, deletion
→ eg:- Binary Search trees

3) Trie → dictionary

4) Network Routing algorithm

Application of Trees in DS

- searching
 - sorting
 - Compiler - validate the syntax.
syntax tree
 - Database design
 - Routers - store the Routing info
 - File system - DS.
- Shortest path among the networks.
- Binary Tree
BST
Balancing Tree
2-3 Trees
TBT
- Dictionaries