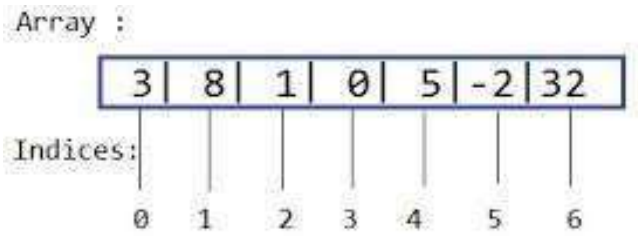
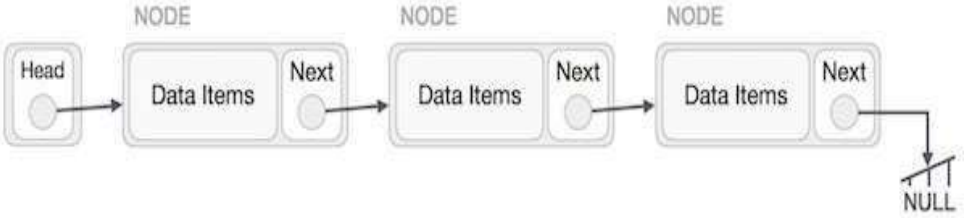
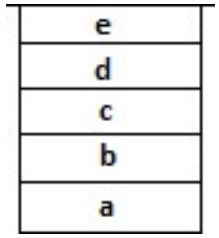
Tree

Unit 6

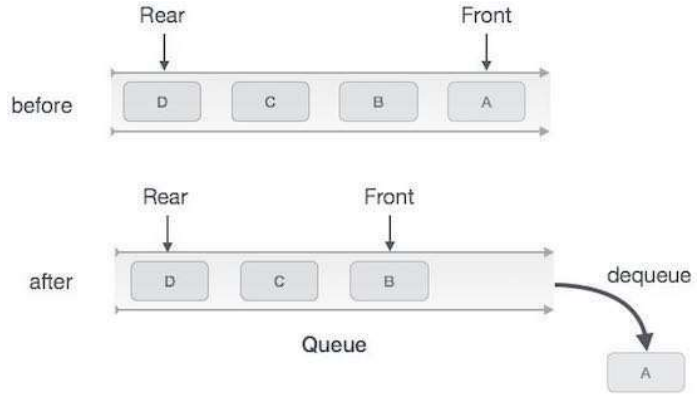
Sofar we discussed Linear data structures like







stack



Queue Dequeue

Introduction to trees

- So far we have discussed mainly linear data structures – strings, arrays, lists, stacks and queues

- Now we will discuss a non-linear data structure called tree.

- Trees are mainly used to represent data containing a hierarchical relationship between elements, for example, records, family trees and table of contents.

- Consider a parent-child relationship

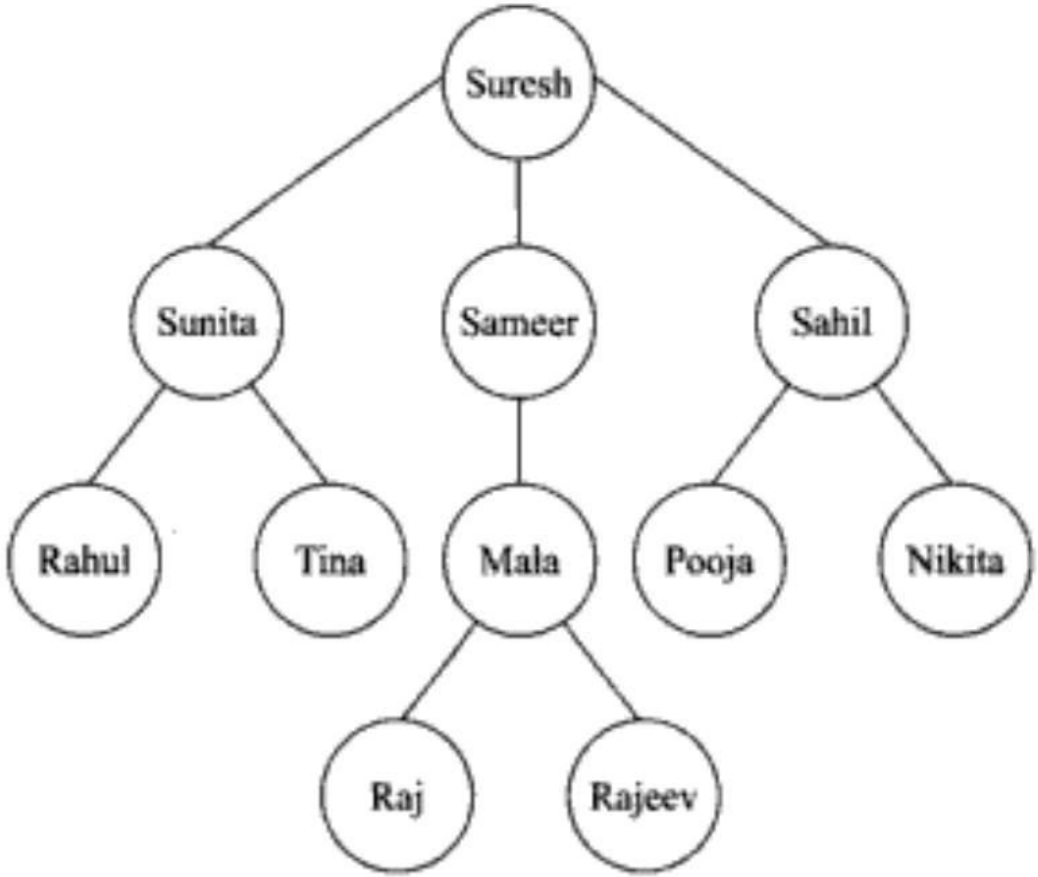
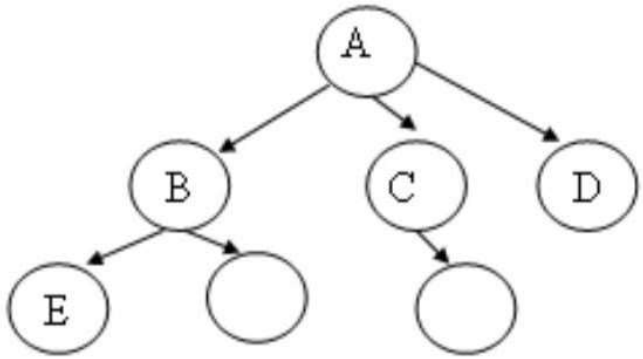


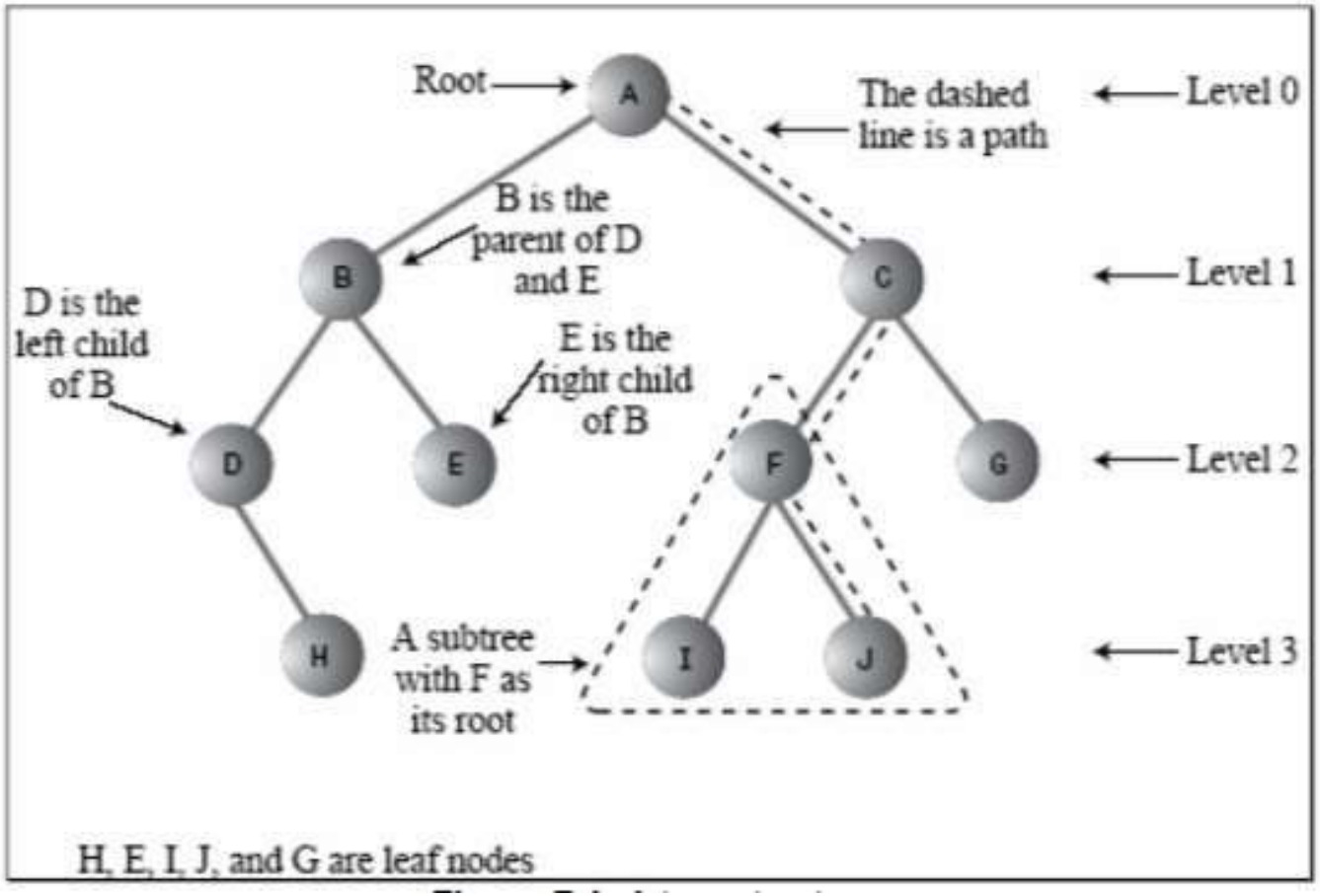
Fig. 8.1 *A Hypothetical Family Tree*

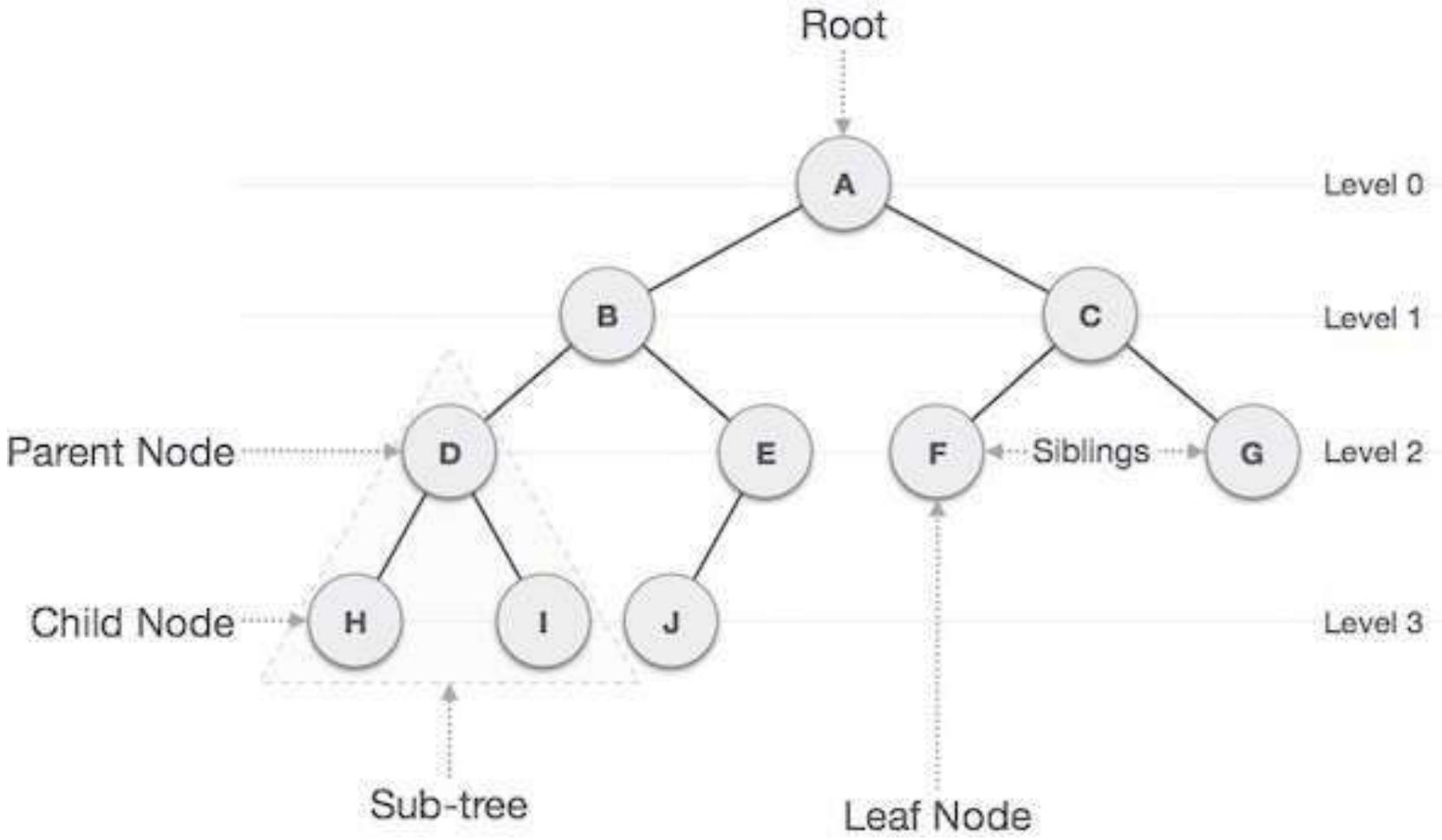
Tree

- A tree is an abstract model of a hierarchical structure that consists of nodes with a parent-child relationship
  - Tree is a sequence of nodes
  - There is a starting node known as a root node
  - Every node other than the root has a parent nod
  - Nodes may have any number of children



A has 3 children, B, C, D
A is parent of B


Root
A
The dashed line is a path
Level 0
B is the parent of D and E
B
C
Level 1
D is the left child of B
E is the right child of B
D
E
F
G
Level 2
H
A subtree with F as its root
I
J
Level 3
H, E, I, J, and G are leaf nodes

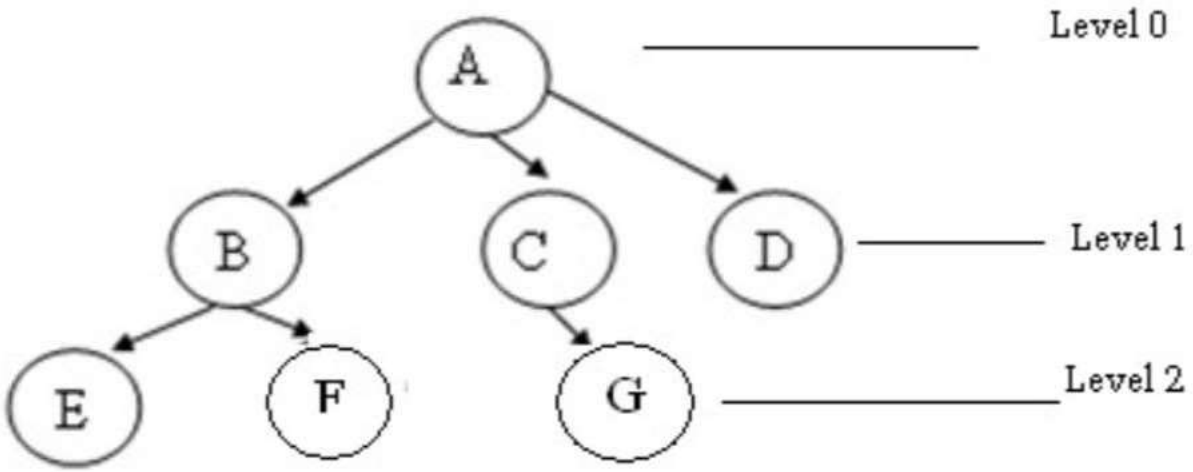

Root
A
B
C
D
E
F
G
H
I
J
Level 0
Level 1
Level 2
Level 3
Parent Node
Siblings
Child Node
Sub-tree
Leaf Node

Some Key Terms:

- Root − Node at the top of the tree is called root.
- Parent − Any node except root node has one edge upward to a node called parent.
- Child − Node below a given node connected by its edge downward is called its child node.
- Sibling – Child of same node are called siblings
- Leaf − Node which does not have any child node is called leaf node.
- Sub tree − Sub tree represents descendants of a node.
- Levels − Level of a node represents the generation of a node. If root node is at level 0, then its next child node is at level 1, its grandchild is at level 2 and so on.
- keys − Key represents a value of a node based on which a search operation is to be carried out for a node.

Some Key Terms:

- Degree of a node:
  - The degree of a node is the number of children of that node
- Degree of a Tree:
  - The degree of a tree is the maximum degree of nodes in a given tree
- Path:
  - It is the sequence of consecutive edges from source node to destination node.
- Height of a node:
  - The height of a node is the max path length form that node to a leaf node.
- Height of a tree:
  - The height of a tree is the height of the root
- Depth of a tree:
  - Depth of a tree is the max level of any leaf in the tree



- ✔ A is the root node
- ✔ B is the parent of E and F
- ✔ D is the sibling of B and C
- ✔ E and F are children of B
- ✔ E, F, G, D are external nodes or leaves
- ✔ A, B, C are internal nodes
- ✔ Depth of F is 2
- ✔ the height of tree is 2
- ✔ the degree of node A is 3
- ✔ The degree of tree is 3

Characteristics of trees

- Non-linear data structure
- Combines advantages of an ordered array
- Searching as fast as in ordered array
- Insertion and deletion as fast as in linked list
- Simple and fast

Application

- Directory structure of a file store
- Structure of an arithmetic expressions
- Used in almost every 3D video game to determine what objects need to be rendered.
- Used in almost every high-bandwidth router for storing router-tables.
- used in compression algorithms, such as those used by the .jpeg and .mp3 file- formats.