

TREES



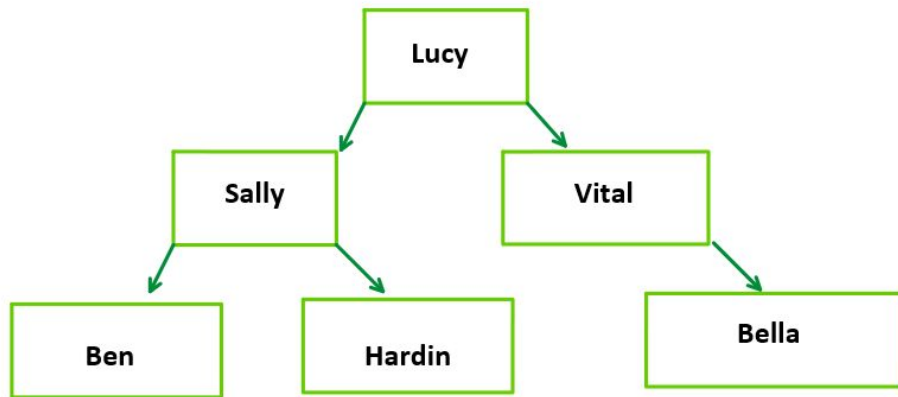
19BQ1A05L1 – Shaik Vaseem Naazleen (CSE-D)

OVERVIEW

- What are trees?
- Terminology
- Binary tree
- Types of binary trees
- Binary search tree
- Traversals
- Applications

WHAT ARE TREES?

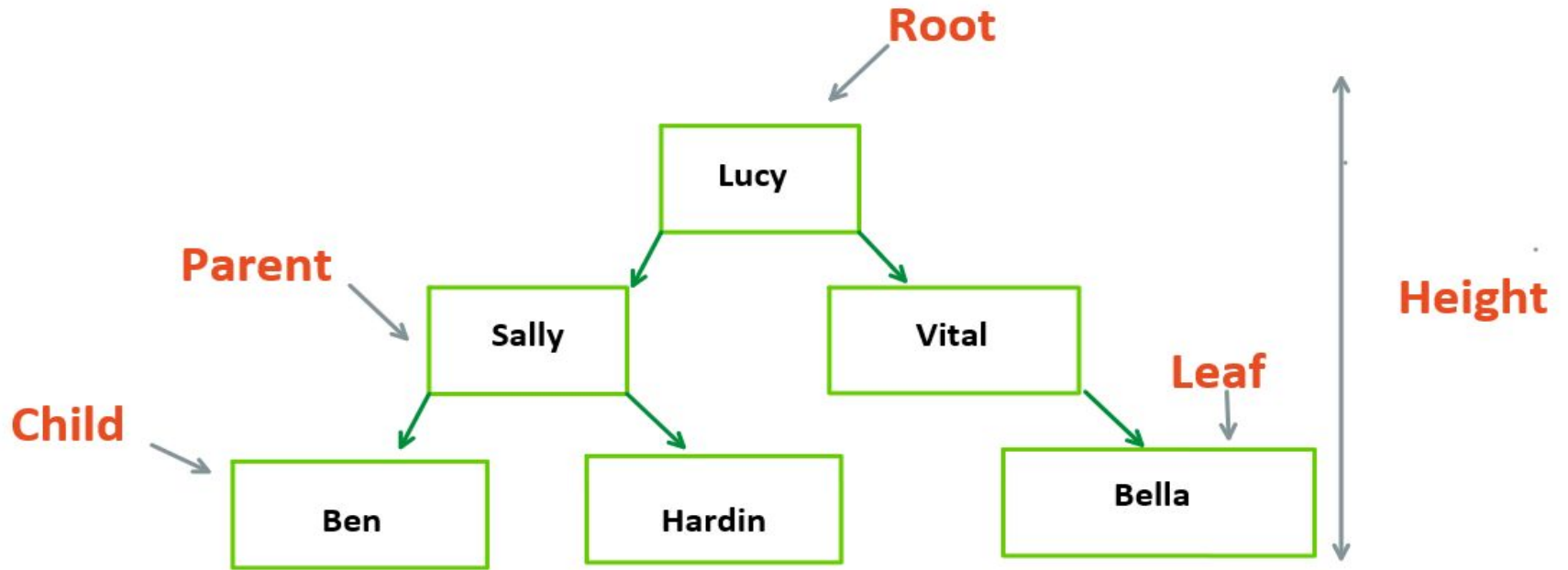
1. Non-linear data structures
2. Mainly used to represent hierarchical relationship between elements
3. parent-child relationship



TERMINOLOGY

1. Node which has no parent is called as **root** node
2. Node which has no children is called as **leaf** node
3. **Parent** is immediate predecessor of a node
4. **child** is immediate successor of a node
5. Maximum number of nodes possible in a path starting from root node to a leaf node is called the **height** of a tree.

TERMINOLOGY

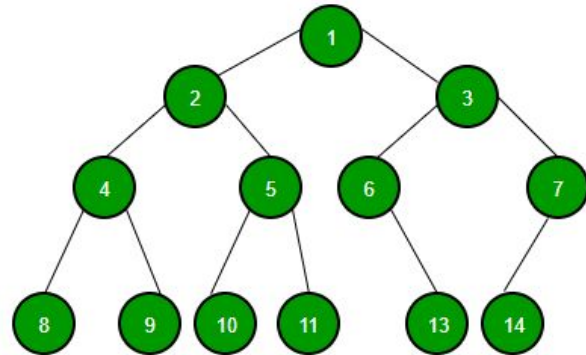


BINARY TREE

A tree whose elements have **at most 2** children is called as a binary tree.

every node consists of 3 components:

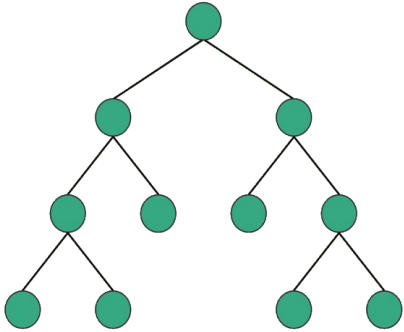
1. Data
2. Left child
3. Right child



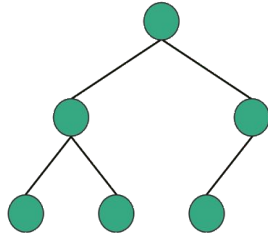
TYPES OF BINARY TREES

1. Full binary tree
2. Complete binary tree
3. degenerate binary tree
4. Perfect binary tree
5. Balanced binary tree

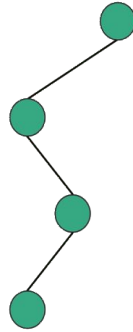
TYPES OF BINARY TREES



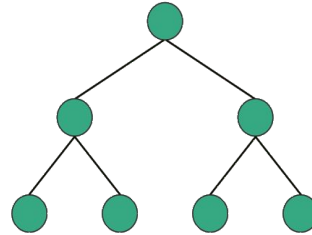
Full



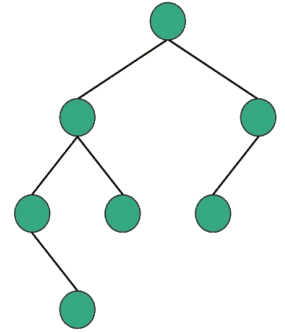
Complete



Degenerate



Perfect

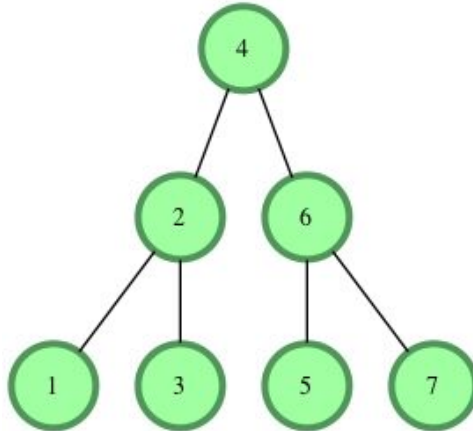


Balanced

BINARY SEARCH TREE (BST)

A binary tree with special property:

- every node in left subtree should consist of node with key's less than root node
- every node in right subtree should consist of node with key's greater than root node
- left & right subtree's should also be BST's.

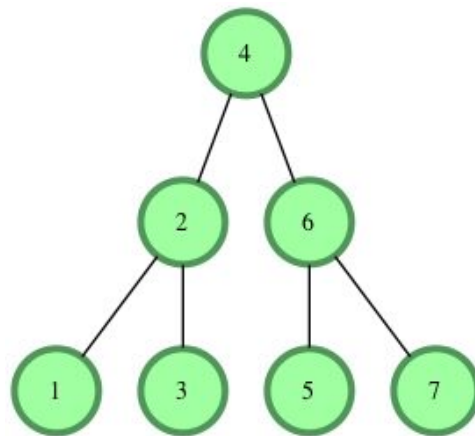


TRAVERSALS

- Inorder traversal
- Preorder traversal
- Postorder traversal
- Level order traversal

INORDER TRAVERSAL

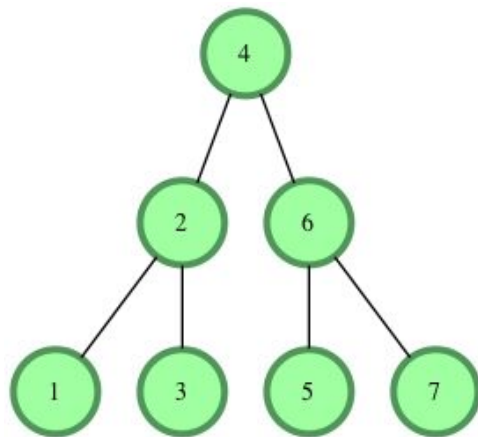
- visit left subtree
 - visit node
 - visit right subtree
- (recursively)



Inorder traversal: 1 2 3 4 5 6 7

PREORDER TRAVERSAL

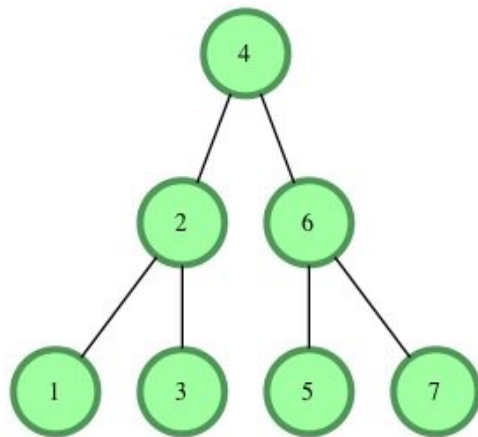
- visit node
 - visit left subtree
 - visit right subtree
- (recursively)



Preorder traversal: 4 2 1 3 6 5 7

POSTORDER TRAVERSAL

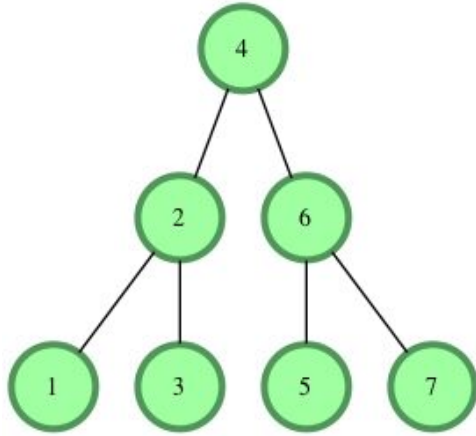
- visit left subtree
 - visit right subtree
 - visit node
- (recursively)



Postorder traversal: 1 3 2 4 5 7 6

LEVEL ORDER TRAVERSAL

visit nodes level by level



Level order traversal : 4 2 6 1 3 5 7

APPLICATIONS

- Represent organization
- Represent computer file system
- Networks to find best path on internet
- Chemical formula representation
- Ordered storage to be used in binary search
- Decision trees
- Encoding

THANK YOU