Main memory : RAM

Dated:

## DATA STRUCTURE:

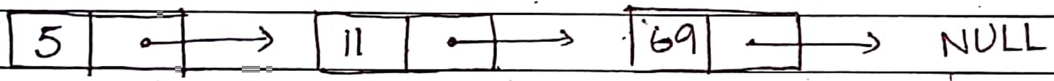
Data structure are the ingredients by which we can make efficient algorithm OR way to arrange data in main memory for efficient usuage.
e.g.: Arrays , List , Linked list , Queue , stack. etc

## ALLGORITHMS:

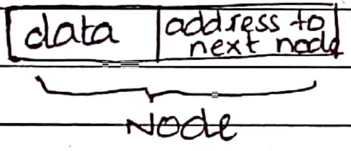
Sequence of steps on data using efficient data structures to solve a given problem.

## LINKED LIST:

It resembles with the concept of an array but it isn't contagious data stored. It has a extra node attach with data which is pointer to the next resembler node.
e.g

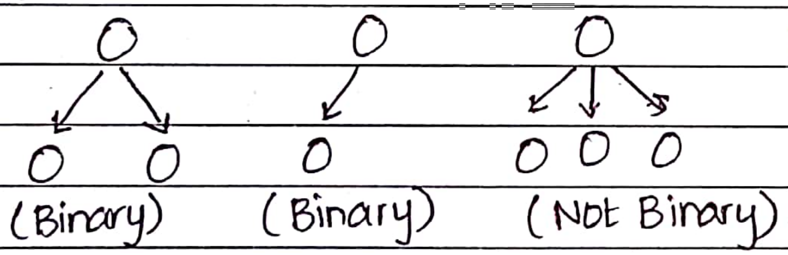


It always end at null.



## BINARY TREE:

Binary tree is a tree which has atmost 2 children for all nodes.



(Ham 2 Hamare 2 :-)
Nodes child will be = 0 , 1 , 2

- * nodes = n
- * edges = n-1
- * Degree = No of direct children (For a node)

* insertion and deletion is as easy as water.

Terminologies for tree

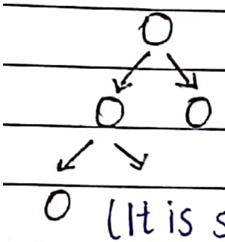
- * Root → Upper most node
- * Parent → Node which connects to child
- * child → Node which is connected to parent
- * leaf: Outermost node
- * Internal: Node with atleast one chil
- * Depth: No of edges from roo
- * Height: No of edges from root to deepest leaf

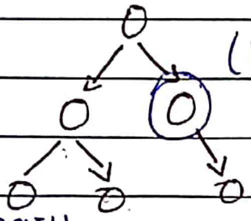
Dated:

## TYPES OF BINARY TREE:

- Full / strict Binary Tree:

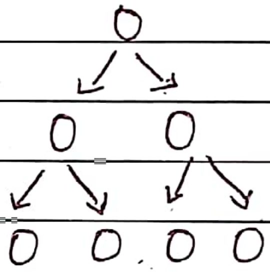
All nodes have either 0 or 2 children


(It is strict binary tree)


(It isn't strict binary tree)

- Perfect Binary Tree:

Internal nodes have 2 children
+ All leaf nodes are on same depth

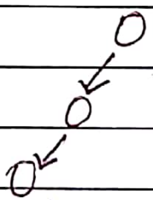


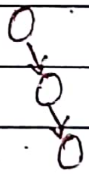
- Complete Binary tree:

All levels are completly filled except possibly the last level + last level must be left allingned

- Degnerate Binary tree

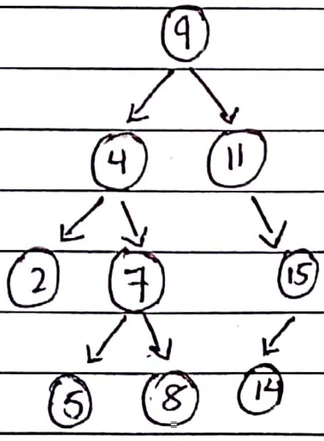
Every Parent node has exactly one child


Left skeewed


Right skeewed

## BINARY SEARCH TREE:

- All nodes of left subtree are lesser
- All nodes of right subtree are greater
- There are no duplicate nodes
- Left and Right are also BST
- In order Traversal of a BST gives an ascending sorted array.


9
4
11
2
7
15
5
8
14