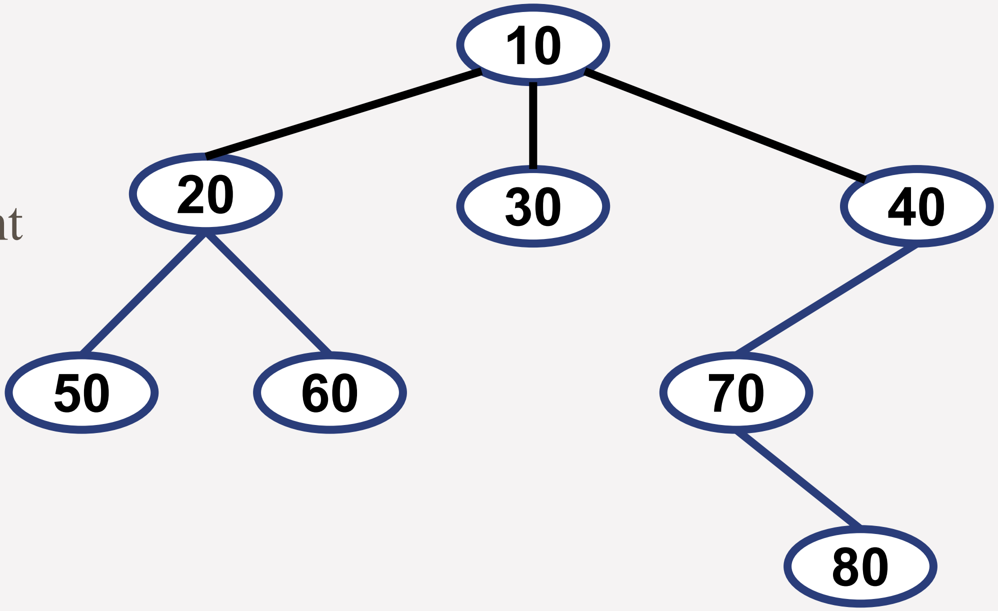
Trees

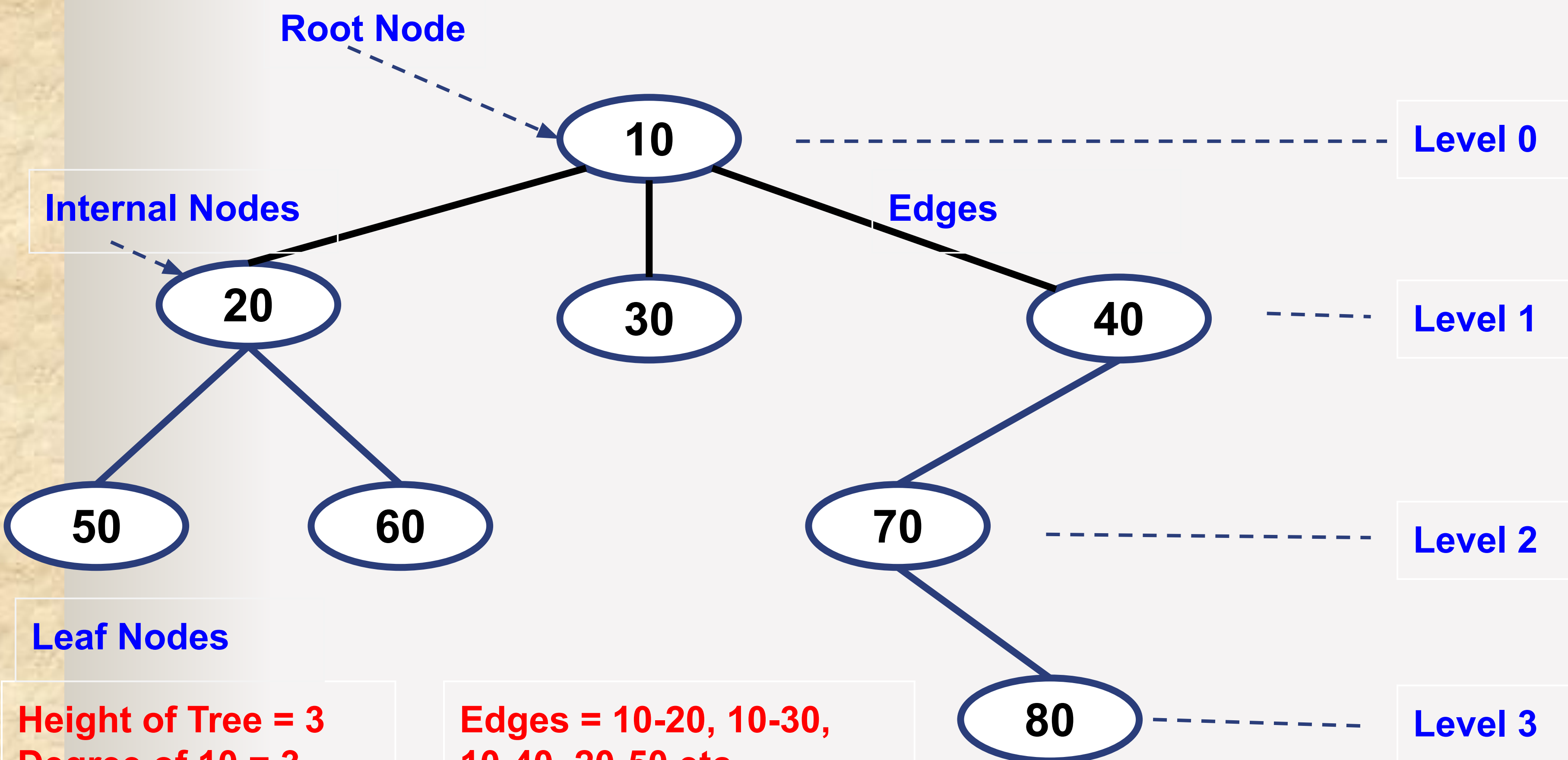
Prof. Harish D.G.
Dept. of Computer and IT
College of Engineering,Pune(COEP)
www.harishgadade.com

Tree

1. Introduction
2. Basic terms/Terminology
   1. Definition of Tree
   2. Node
   3. Degree
   4. Leaf Nodes
   5. Interior Nodes
   6. Children and Parent
   7. Siblings
   8. Degree of tree
   9. Level
   10. Height of tree



Tree



Height of Tree = 3
Degree of 10 = 3
Degree of 40 = 1
Degree of Tree = 3

Edges = 10-20, 10-30, 10-40, 20-50 etc

Height of Tree = 3

Internal Nodes = 20, 40, 70
Leaf Nodes = 50, 60, 30, 80

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

A tree is a DS with set of one or more nodes. There is a special node called root and remaining nodes are partitioned into disjoint groups T1,T2….Tn. Where Ti- sub tree

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Each item in a tree called Node

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Number of subtrees of a node in a given tree

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Maximum degree of node in a given tree

Tree

1. Introduction
2. Basic terms/Terminology
   1. Definition of Tree
   2. Node
   3. Degree
   4. Leaf Nodes
   5. Interior Nodes
   6. Children and Parent
   7. Siblings
   8. Degree of tree
   9. Level
   10. Height of tree

A node with degree zero

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Any Node except root node whose degree not zero

Tree

1. Introduction
2. Basic terms/Terminology
   1. Definition of Tree
   2. Node
   3. Degree
   4. Leaf Nodes
   5. Interior Nodes
   6. Children and Parent
   7. Siblings
   8. Degree of tree
   9. Level
   10. Height of tree

Root of sub tree is parent & sub tree is its children

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Children's of the same parents

Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree of Node
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Maximum degree of nodes in a given tree

Tree

1. Introduction
2. Basic terms/Terminology
   1. Definition of Tree
   2. Node
   3. Degree of Node
   4. Leaf Nodes
   5. Interior Nodes
   6. Children and Parent
   7. Siblings
   8. Degree of tree
   9. Level
   10. Height of tree

Root is at level 0
If node is at level ‘n’, then its children’s are at level n+1

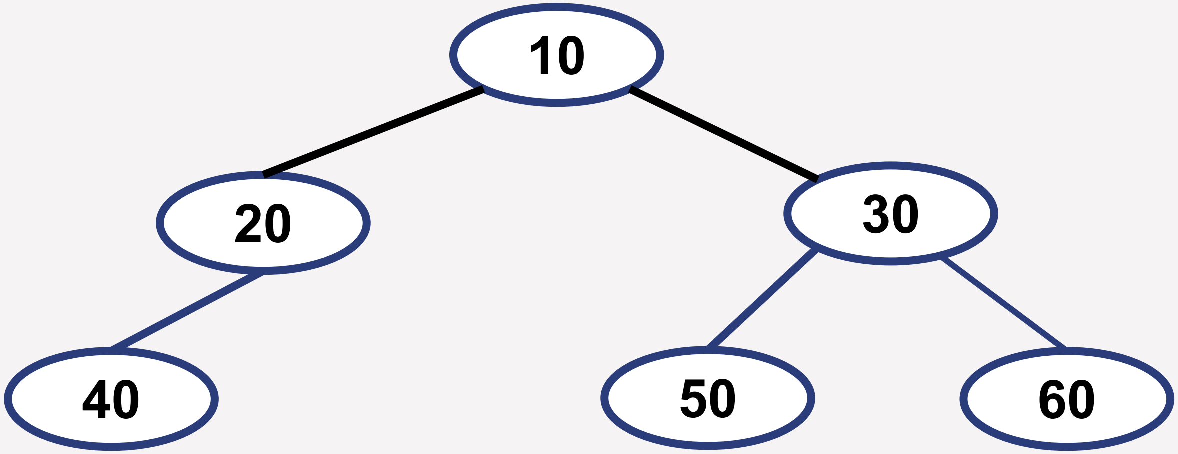
Tree

1. Introduction
2. Basic terms/Terminology
    1. Definition of Tree
    2. Node
    3. Degree of Node
    4. Leaf Nodes
    5. Interior Nodes
    6. Children and Parent
    7. Siblings
    8. Degree of tree
    9. Level
    10. Height of tree

Maximum level of any node in a given tree

Binary Tree

- Definition
- Properties of Binary tree
- Types of Binary Tree
    1. Skewed Binary tree
    2. Complete Binary Tree
    3. Full Binary Tree



Binary Tree

- Definition
- Properties of Binary tree
- Types of Binary Tree
  1. Skewed Binary tree
  2. Complete Binary Tree
  3. Full Binary Tree

Properties of BT:

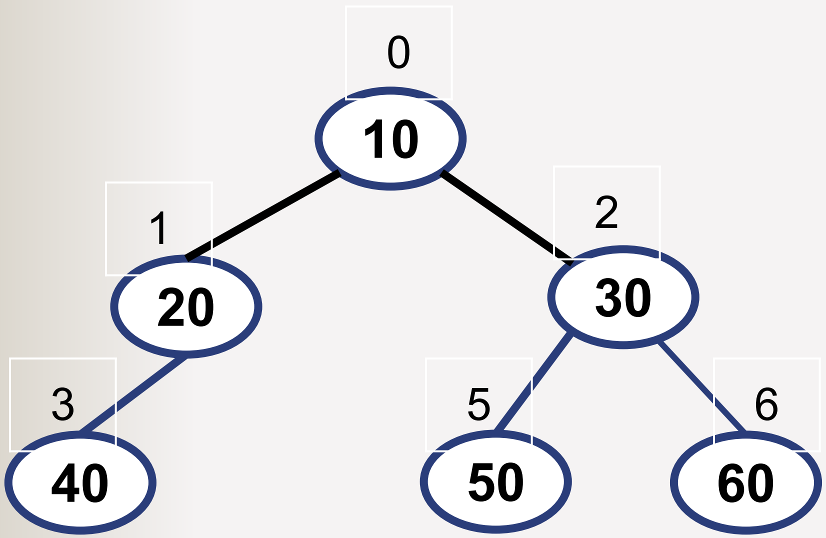
1. For Binary Tree, maximum number of nodes at level L are 2^L
2. A full Binary tree of height 'h' has **($2^{h+1} - 1$)** total nodes
3. Total number of external nodes in a binary tree are internal nodes + 1.

   i.e. E = I + 1

Rules:

- Parent (i) = floor(i-1)/2
- Leftchild (i) = (2i+1)
- Rightchild(i)= (2i + 2)

Binary Tree



Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

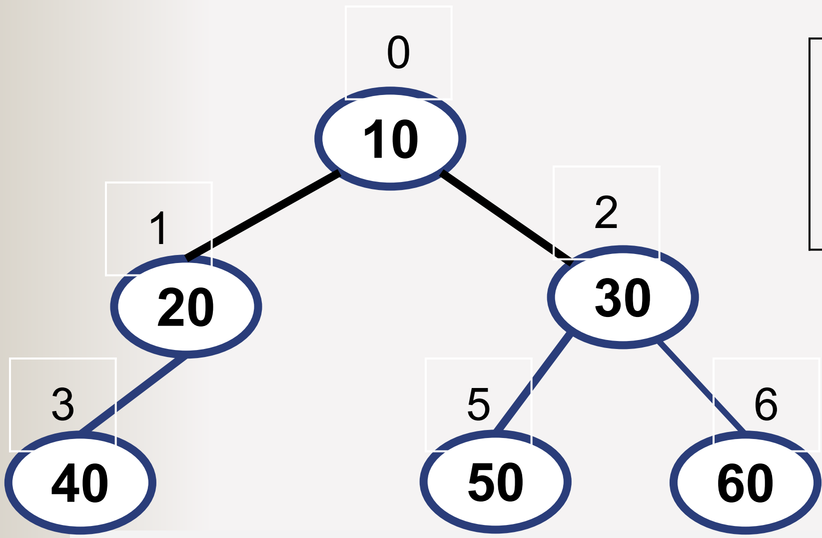
Properties of BT:

1. For Binary Tree, maximum number of nodes at level L are **2^L**
2. A full Binary tree of height ‘h’ has **($2^{h+1} - 1$)** total nodes
3. Total number of external nodes in a binary tree are internal nodes(Including Root) + 1.

   i.e. E = I + 1

E = I + 1
E = 3 + 1 = 4

Array Representation Binary Tree



Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

E = I + 1
E = 3 + 1 = 4

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

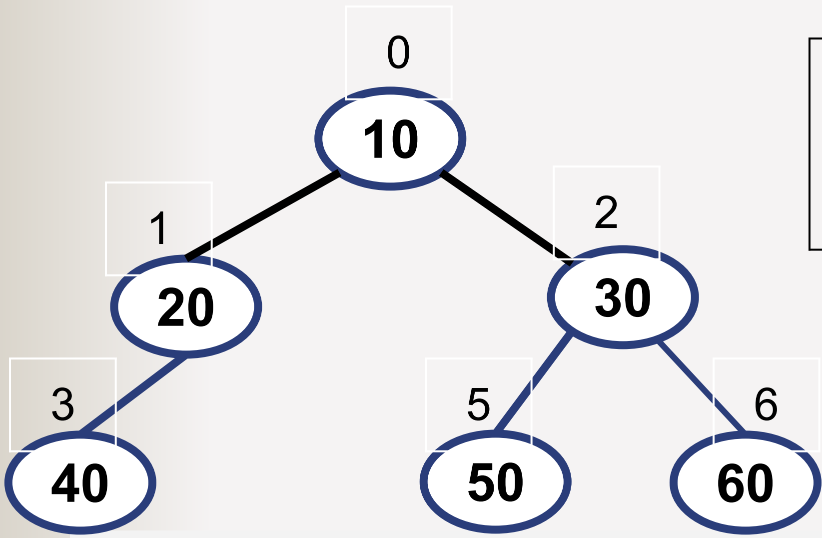
0	1	2	3	4	5	6
10	20	30	40		50	60

Rules:

- Parent (i) = floor(i-1)/2
- Leftchild (i) = (2i+1)
- Rightchild(i)= (2i + 2)

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Array Representation Binary Tree



0	1	2	3	4	5	6
10	20	30	40		50	60

Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

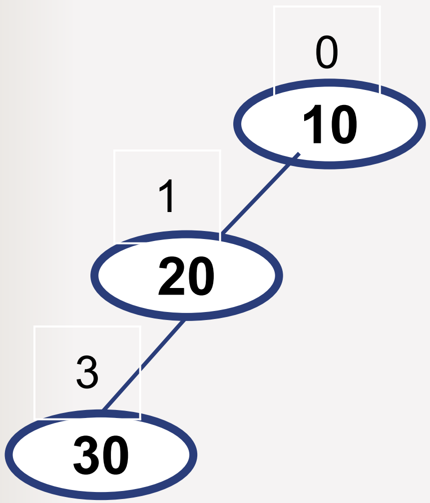
E = I + 1
E = 3 + 1 = 4

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Parent(1) = Floor(0/2) = 0
Leftchild(1) = (2*1+1) = 3
Rightchild(1) = (2*1+2) = 4

Array Representation Binary Tree



Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

E = I + 1
E = 3 + 1 = 4

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

0	1	2	3	4	5	6
10	20		30			

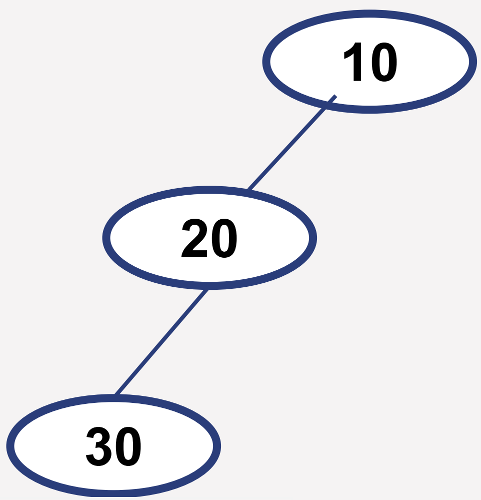
Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Parent(1) = Floor(0/2) = 0
Leftchild(1) = (2*1+1) = 3
Leftchild(1) = (2*1+2) = 4

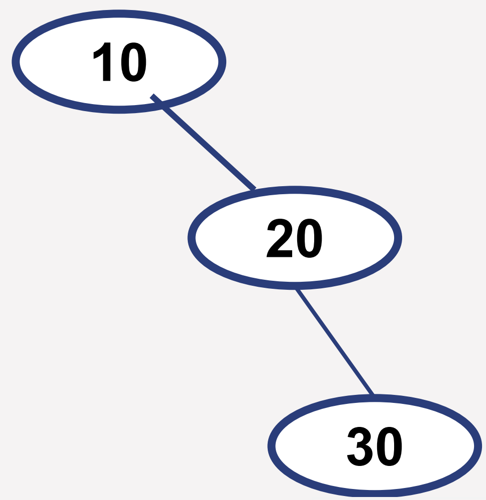
Binary Tree

- Definition
- Properties of Binary tree
- Types of Binary Tree
    1. Skewed Binary tree

Every node is having either only left or right subtree



Left Skew Tree

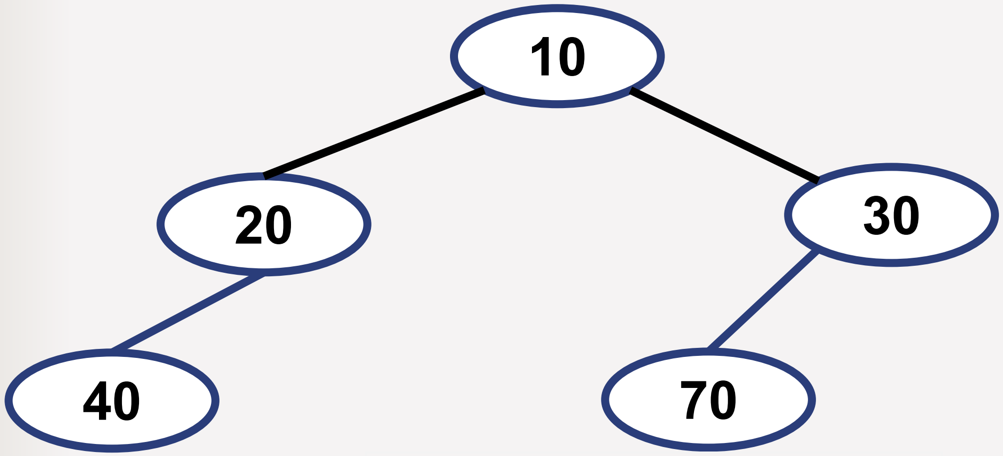


Left Skew Tree

Binary Tree

- Definition
- Properties of Binary tree
- Types of Binary Tree
  1. Skewed Binary tree
  2. Complete Binary Tree

All Leaves are at the same depth / level

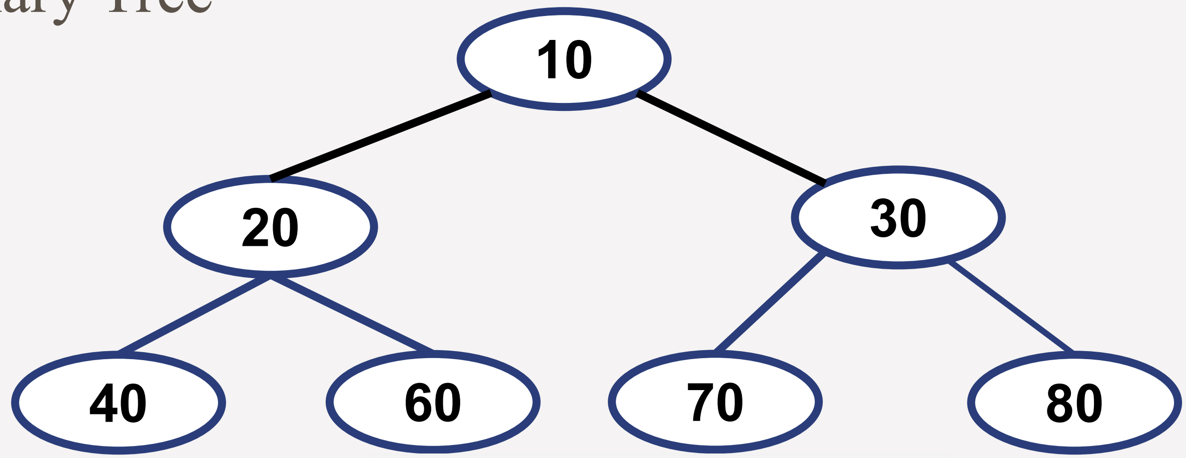


Complete Binary Tree

Binary Tree

- Definition
- Properties of Binary tree
- Types of Binary Tree
    1. Skewed Binary tree
    2. Complete Binary Tree
    3. Full Binary Tree

Every node has zero or two children.
- Also called Strictly B.T.

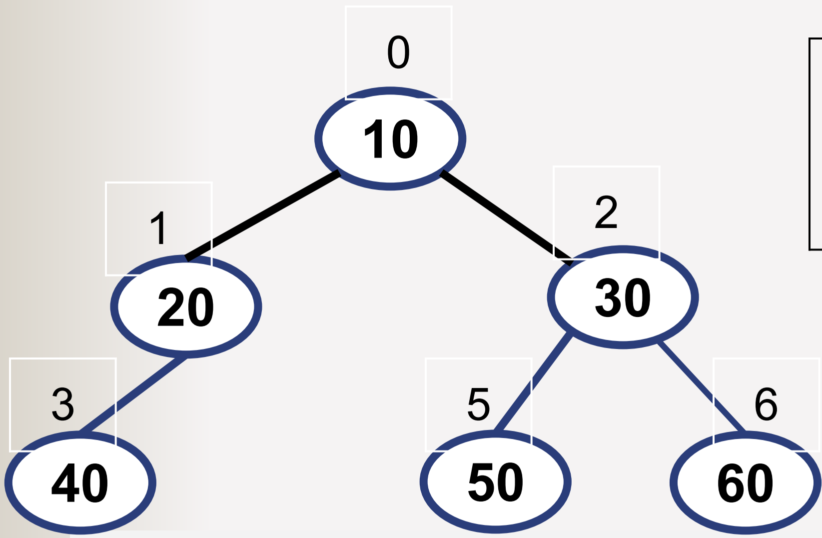


Full Binary Tree

Representation of Binary Tree

1. Sequential / Array Representation
2. Linked Representation
    - Structure Representation
    - Node Representation

Sequential / Array Representation Binary Tree



$$\text{Level } 0 = 2^0 = 1$$
$$\text{Level } 1 = 2^1 = 2$$
$$\text{Level } 2 = 2^2 = 4$$

$$E = I + 1$$
$$E = 3 + 1 = 4$$

$$\text{Total Nodes} = 2^{(h+1)} - 1$$
$$= 2^3 - 1 = 7$$

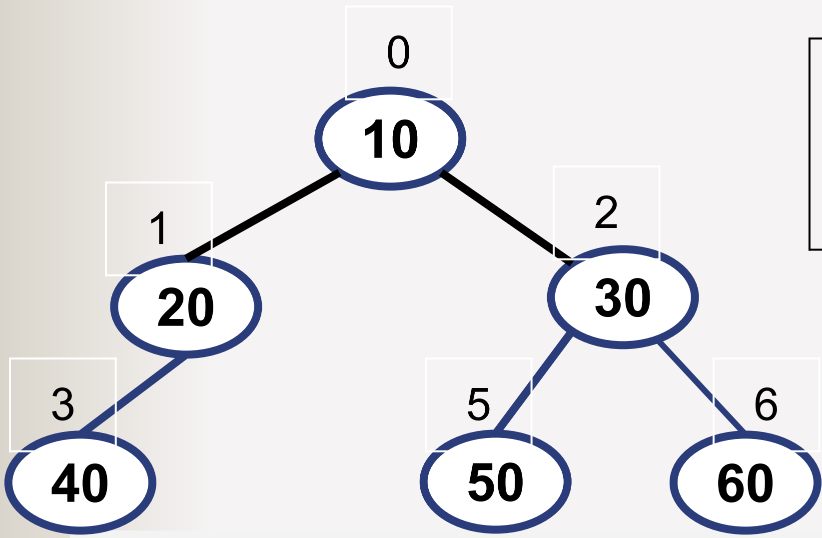
0	1	2	3	4	5	6
10	20	30	40		50	60

Rules:

- Parent (i) = floor(i-1)/2
- Leftchild (i) = (2i+1)
- Rightchild(i)= (2i + 2)

$$\text{Total Nodes} = 2^{(h+1)} - 1$$
$$= 2^3 - 1 = 7$$

Sequential / Array Representation Binary Tree



Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

E = I + 1
E = 3 + 1 = 4

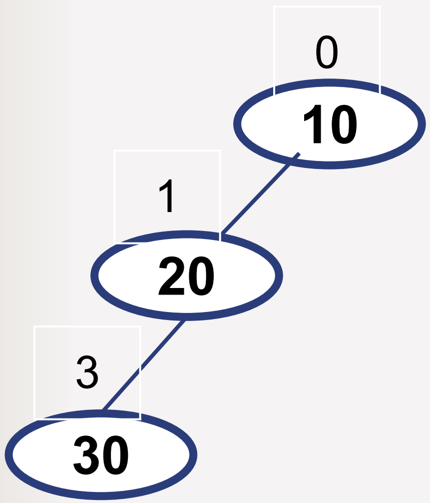
Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

0	1	2	3	4	5	6
10	20	30	40		50	60

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Parent(1) = Floor(0/2) = 0
Leftchild(1) = (2*1+1) = 3
Leftchild(1) = (2*1+2) = 4

Sequential / Array Representation Binary Tree



Level 0 = 2^0 = 1
Level 1 = 2^1 = 2
Level 2 = 2^2 = 4

E = I + 1
E = 3 + 1 = 4

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

0	1	2	3	4	5	6
10	20		30			

Total Nodes = $2^{(h + 1)} - 1$
= $2^3 - 1 = 7$

Parent(1) = Floor(0/2) = 0
Leftchild(1) = (2*1+1) = 3
Leftchild(1) = (2*1+2) = 4

Linked Representation of Binary Tree

Structure Representation:

```
struct node
{
      struct node *LC;
      int data;
      struct node *RC;
};
```

*LC	Data	*RC

Node Structure

Linked Representation of Binary Tree

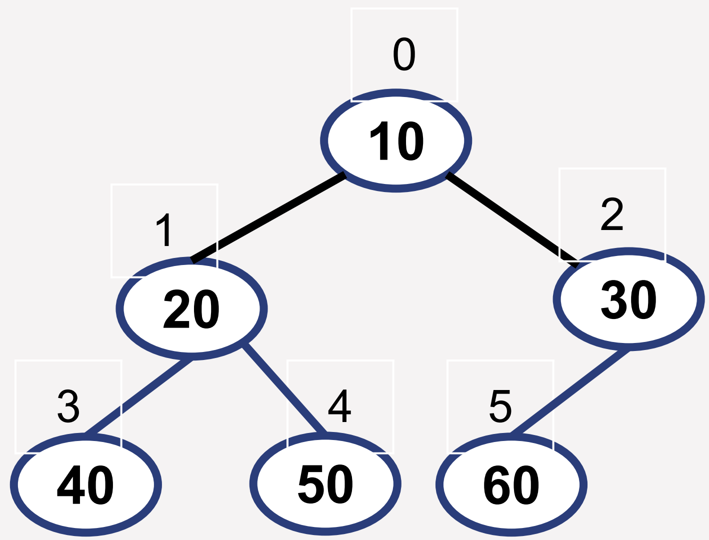
Structure Representation:

```
struct node
{
        struct node *LC;
        int data;
        struct node *RC;
};
```

*LC	Data	*RC

Node Structure

e.g. 10,20,30,40,50



Linked Representation of Binary Tree

e.g. 10,20,30,40,50, 60