The Shirpur Education Society's
R. C. Patel College of Engineering & Polytechnic, Shirpur

Unit – VI TREES

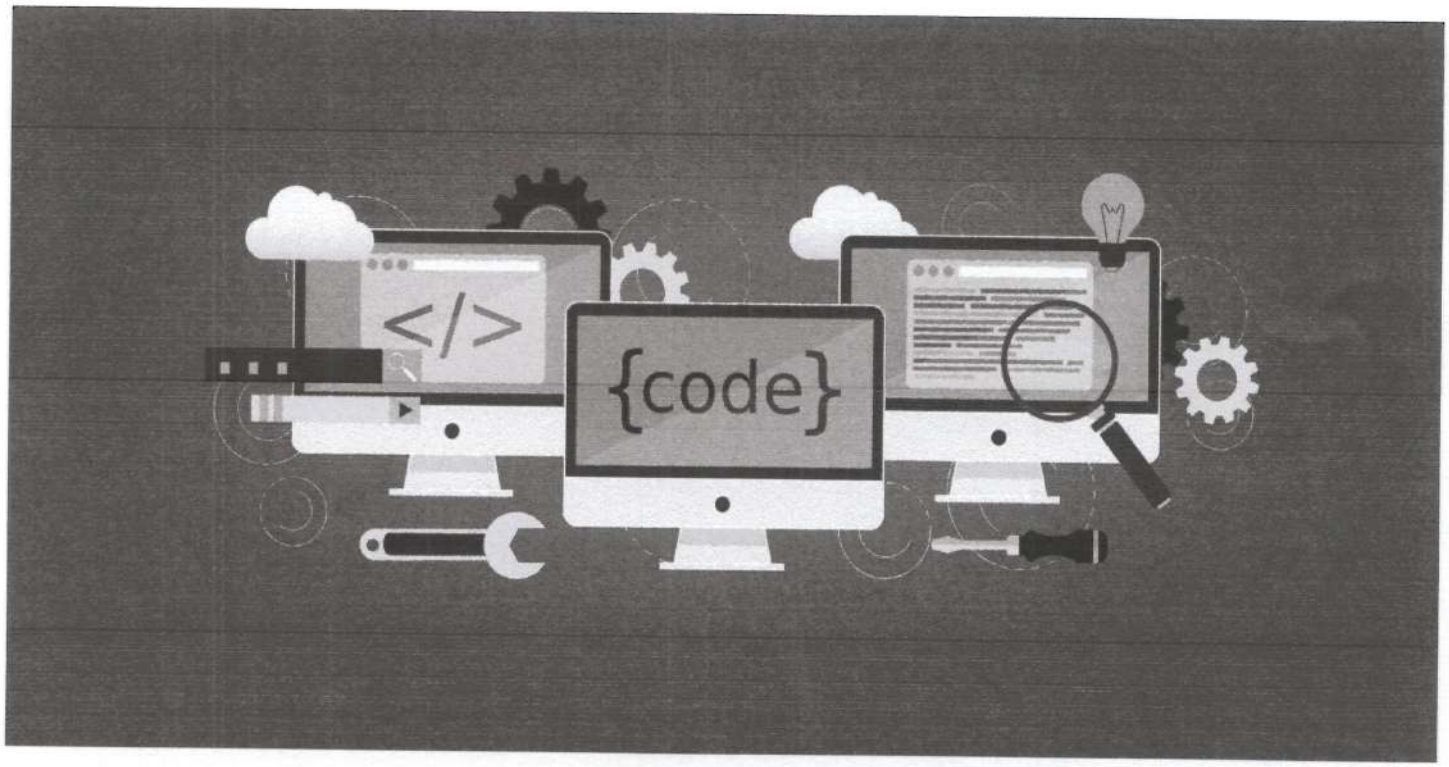
Course Title–Data Structure using C (DSU)

Course Code - 313301

Programme Name - Computer & Computer Science and Engineering

Semester– Third (2nd Year)

Prepared By: Mrs. M.A.Gulhane



No. of pages :- 16

Unit :- 6 Tree

Introduction to Tree :

- Tree is a data structure in which one node is connected with several node and in turns these several nodes are connected with several other nodes.
- Tree data structure enhances the flexibility in traversing and is very powerful & flexible data structure.
- Use of tree is representation of data which contains hierarchical realionship between elements for e.g. records, family trees & table of contents.

Defination :- A tree may be defined as a finite set 'T' of one or more nodes such that there is a node designated as the root of the tree and the other nodes are divided into $n \geq 0$ disjoint sets $T_1, T_2, T_3, T_4 \ldots T_n$ are called the sub trees or childrens of the root.

Advantages of Tree :

- Tree reflect structural relationship in between the records of data.
- Hierarchies can be represented by trees.
- Insertion & Searching can be done efficiently in trees.
- Trees are very flexible DS, allowing to move sub-trees around with minimum efforts.

Tree Terminologies:

1) Degree of a Node: The total number of child nodes of any particular node is called as degree of Node.

2) Node: Every Element in the tree is called as Node which contains data & links to other nodes.

3) Root: The node which is placed at the top of tree is called root node. Every tree has only one root node

4) Leaf node: The lower nodes which does not have any child node are called as leaf node.

5) Degree of a Tree: Degree of tree is the degree of node/nodes in the tree having maximum degree.

6) Level of a node: The level of root node is considered as 0(zero). the level of its child will be One and so on.
The level of any node is one more than its parent.

7) Depth / Height of Tree: The maximum level of any leaf node in the tree is called depth or height of the tree.

8) Edge: The connecting link between any two nodes is called as edge.

9) path:- Sequence of edges from one node to another is called as path between those two nodes.

10) parent: The node which has child node is called as parent. This node is predecessor.

11) child: The node which has a link from its parent node is known as child node. This node is successor node.

12) Ancestor: The ancestors of a node are all the nodes along the path from the root to that node.

13) In-Degree:- The number of edges coming into that node from its parent is in-degree.

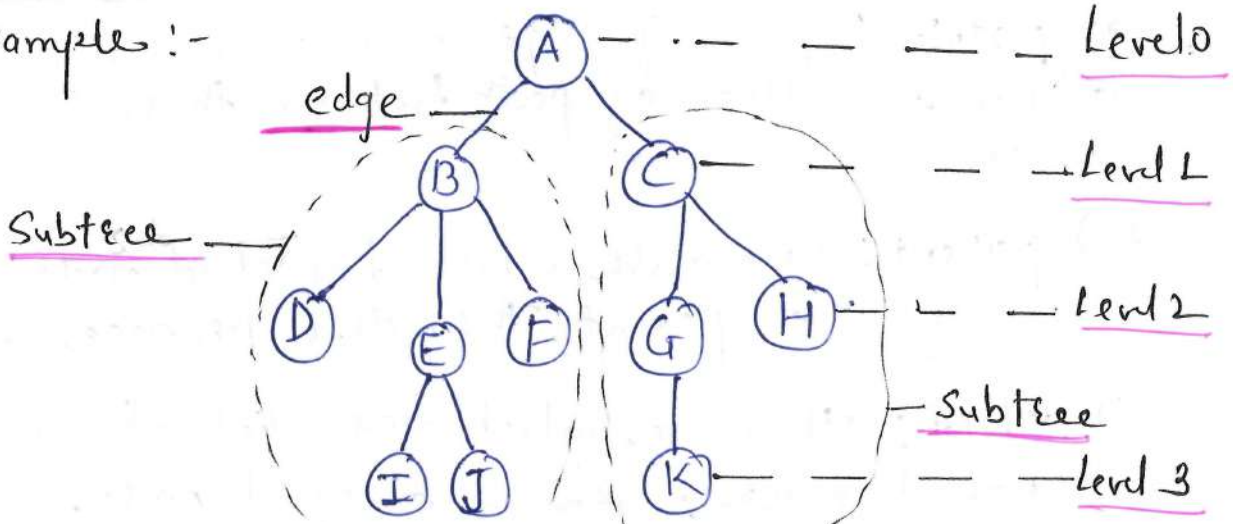
14) Out-degree:- The out-degree of a node is the number of children that node has.

15) Siblings: The node which belongs to same parents are known as siblings.

16) Descendant Node: Descendents of a node are all such nodes in downwards direction which are reachable from that node.

17) SubTree: In a tree, every child from a node forms a subtree recursively. Every child node will form a subtree on its parent node

Example :-



1) Nodes :- for given Tree are A, B, C, D, E, F, G, H, I, J and K are nods.

2) Roots :- A.

3) Leaf nodes : D, I, J, F, K and H.

4) Degree of node : Degree of B is 3
Degree of A is 2
Degree of F is 0.

5) Degree of Tree : Tree of B is maximum
i.e. 3. Deg. of Tree is 3

6) Level of Node :- shown on diagm.

7) Depth / Height of Tree :- Depth / Height is 3.

8) Edge : shown on diagram.

9) path : path between A & J is A-B-E-J.
— " — C & K is C-G-K.

10) parent : · A, B, C, E, G are parent node

11) child : B & C are children of A
G. & H are children of C., K is
K is children of G.

12) Ancestor : B & A are ancestor of E.
C & A are ancestor of G.

13) Descendents : G, H, and K are descendents
of node C.

14) Siblings : B & C are siblings
D, E, & F are siblings
G & H are siblings.
I & J are siblings.

15) subTrees : shown. In diagram.

Types of Tree :- 1) General Tree
2) Binary Tree
3) Binary Search Tree.

1) General Tree : General Tree T is a finite set of one or more nodes such that there is one designated node r, called the root of T, & remaining nodes are partitioned into n>, 0 disjoint subset $T_1, T_2, \ldots T_n$, each of which is tree, & whose roots $r_1, r_2, \ldots r_n$ respt. are children of r.

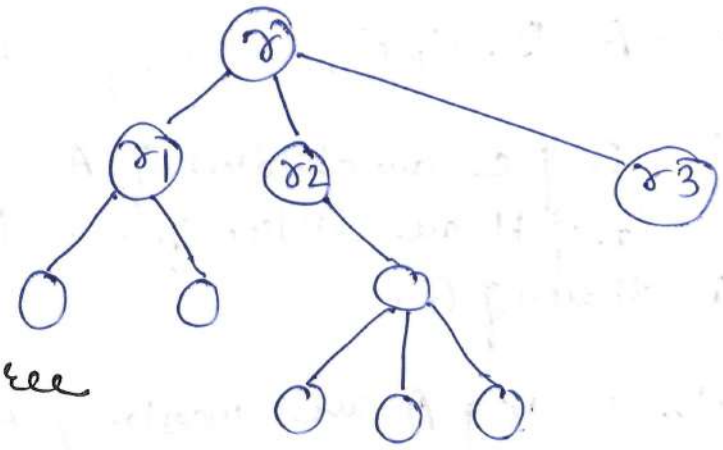


fig:
General Tree

2) Defination: A tree in which every node can have maximum of two child nodes is known as Binary Tree. (Binary Tree)

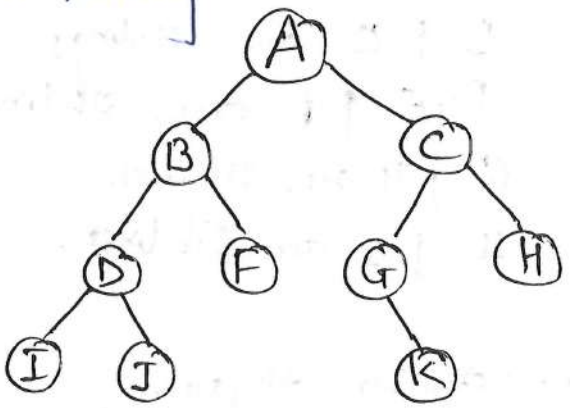


fig: Binary Tree.

Types of Binary Tree :-

1) Strictly Binary Tree: A binary tree in which every node has either two or zero number of child nodes is called Strictly Binary Tree.

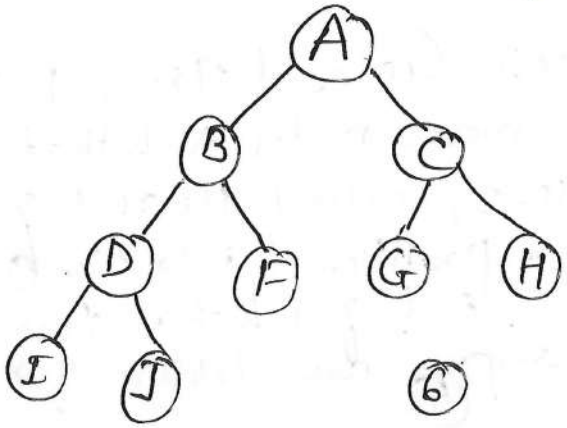


fig: Strictly Binary Tree.

2) Complete Binary Tree: A binary Tree in which every internal node has exactly two child nodes and all the leaf nodes are at same level is called Complete Binary Tree.

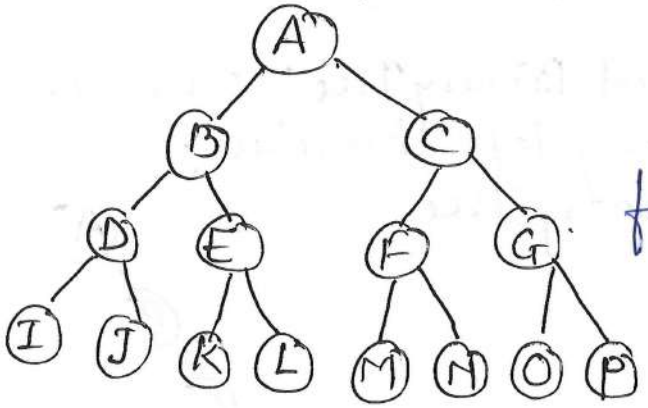


fig: Complete Binary Tree.

3) Almost Complete Binary Tree: A binary tree with L Levels, having all level 1 to L-1 are completely filled (without gap) or any gaps and last level L is partially filled from left to right, is known as Almost Complete Binary Tree

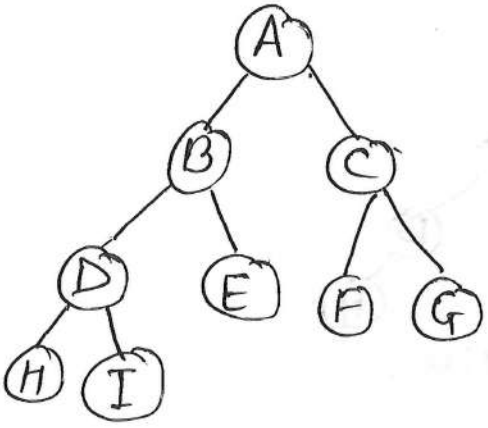


fig: Almost Complete Binary Tree.

4) Skewed Binary Tree : The Binary Tree in which either only left branches are present or only right branches are present is called as skewed Binary Tree.

— Left skewed Binary Tree: The skewed binary tree having only left branches is known as Left skewed Binary Tree.

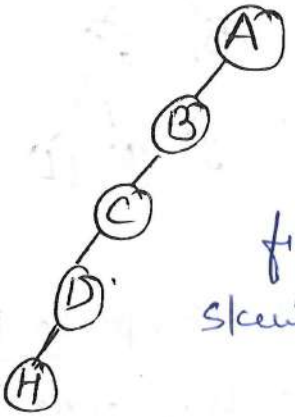


fig: Left skewed Binary Tree.

— Right skewed Binary Tree :- The skewed binary tree having only right branches is known as Right skewed Binary Tree.

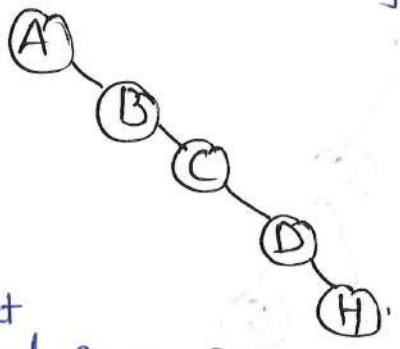


fig: Right skewed Binary Tree.

5) Extended Binary Tree: The full binary tree obtained by adding dummy nodes to a binary tree is called as Extended Binary Tree

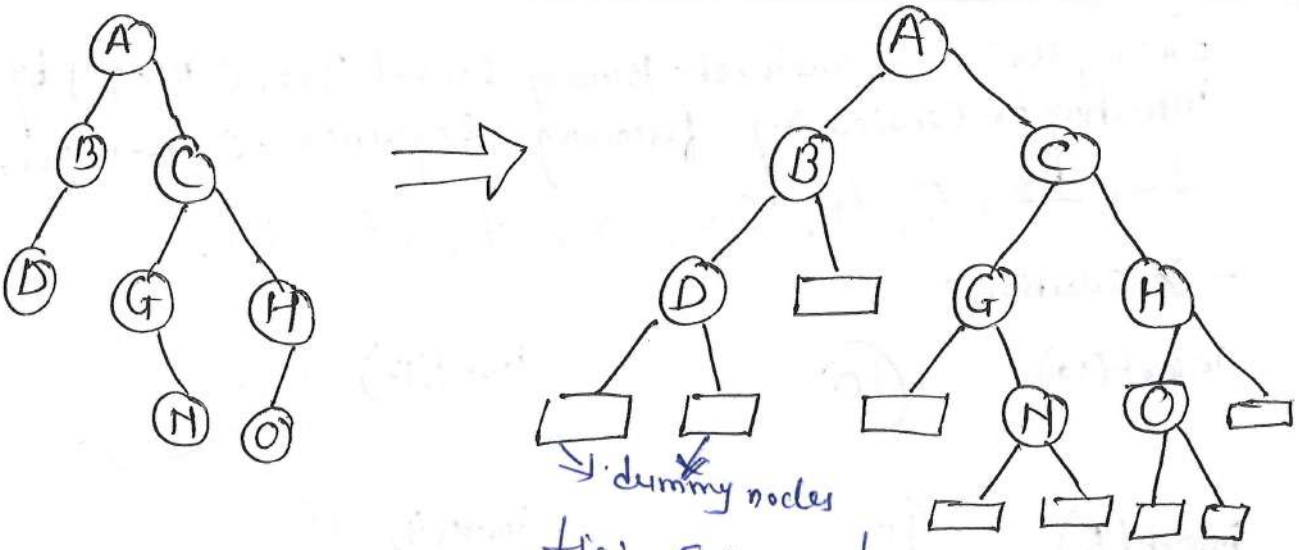


fig:- Extended Binary Tree

Binary Tree Traversal.

Binary Search Tree :- A binary tree in which the data of all the nodes in the left sub-tree of the root node is less than the data of the root and the data of all the nodes in the right sub-tree of the root node is more (greater) than the data of the root, is called as Binary Search Tree.

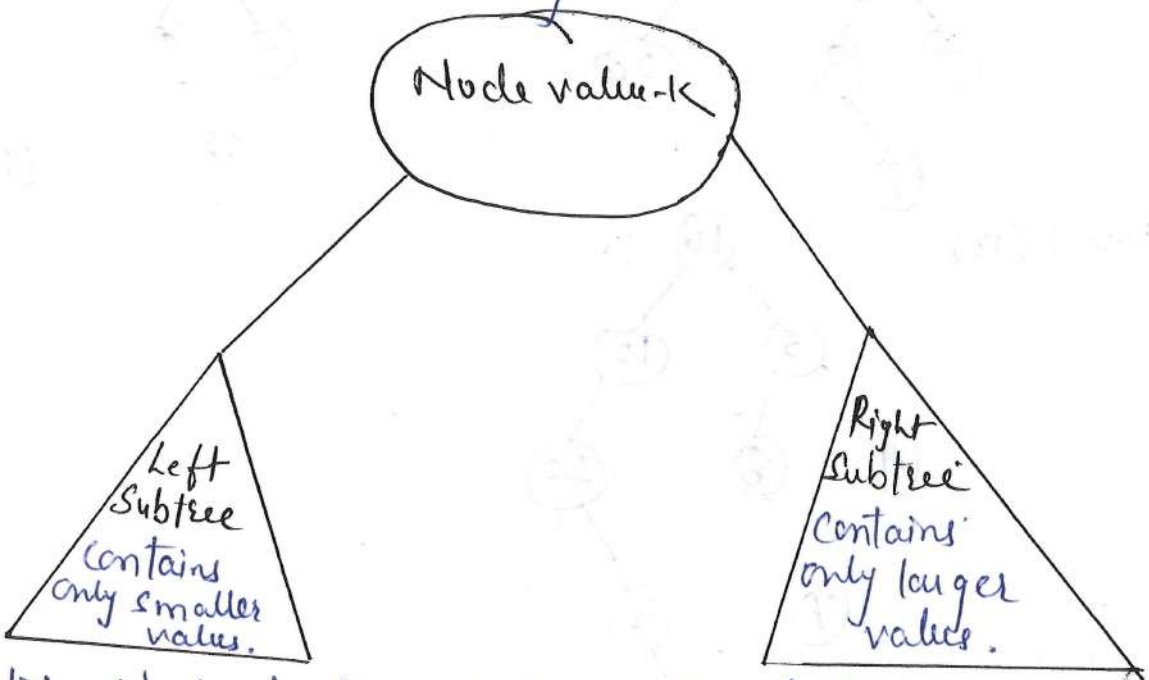


fig:- Node structure of Binary Search Tree.

Examples:- Construct Binary Search Tree (BST) by inserting or considering following sequence of number.

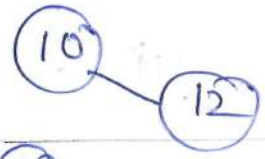
10, 12, 5, 4, 20, 8, 7, 15, 13.

→ solution :-

insert(10)

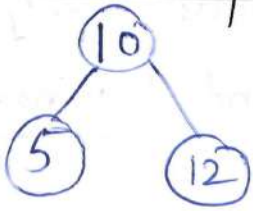
step-1

insert(12)



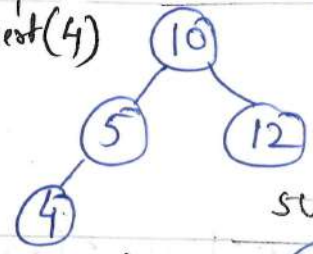
step-2

insert(5)



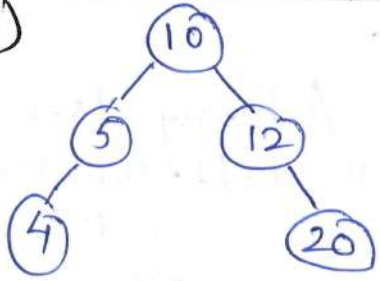
step-3

insert(4)



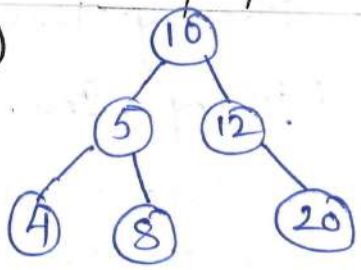
step-4

insert(20)



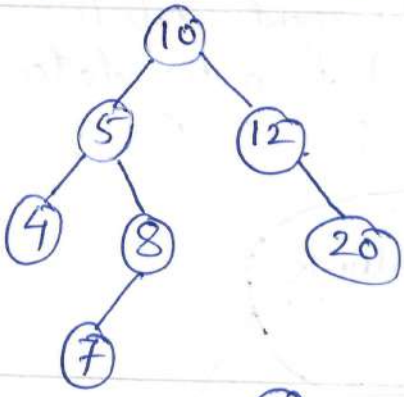
step-5

insert(8)



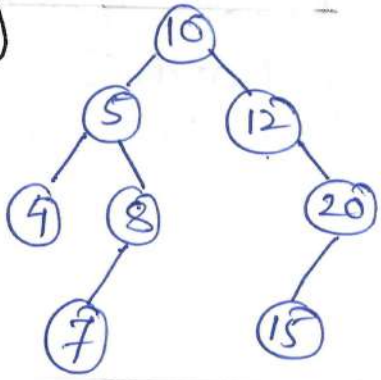
step-6

insert(7)



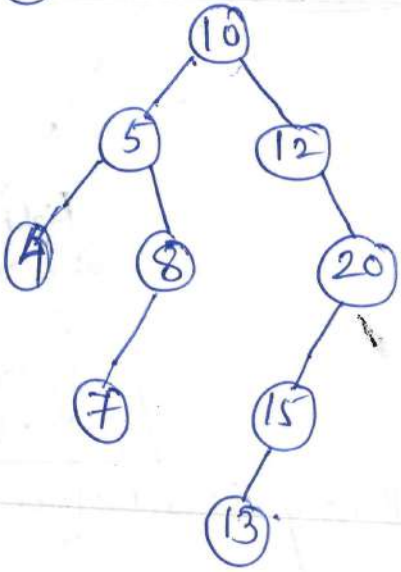
step-7

insert(15)



step-8

insert(13)



step-9

Binary Tree Traversal :-

- Traversing a tree means visiting each node in the tree once. For performing various operation on tree, we have to traverse the tree.
- There are 3 different ways to traverse a tree.
  1) Inorder Traversal
  2) Preorder Traversal
  3) postorder Traversal

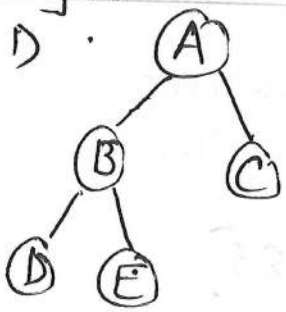
1) Inorder Traversal :- In Inorder traversal node are visited in the following order.

Left subtree → Root → Right subtree.

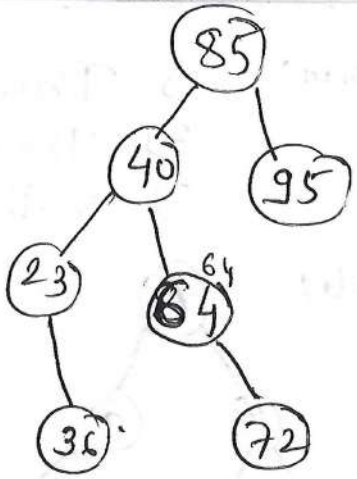
Algorithm :- 1) Traverse the left subtree.
2) Visit the root node
3) Traverse the right subtree.

E.g :-

1)



Inorder sequence is
D - B - E - A - C.
or
(D, B, E, A, C)



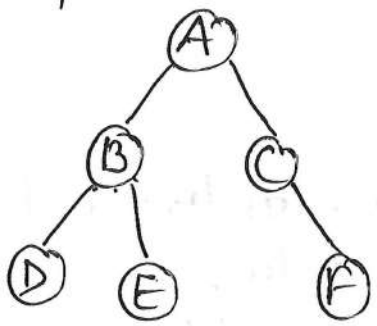
Inorder sequence is
23 - 36 - 40 - 64 - 72 - 85 - 95

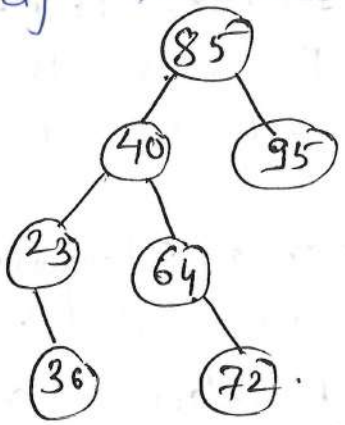
2) Preorder Traversal: In preorder sequence nodes are visited in following way

Root → Left Subtree → Right Subtree.

Algorithm :- 1) Visit the root
2) Traverse the leftsub-tree.
3) Traverse the Right Sub-tree.

Example :-





Preorder Sequence → A - B - D - E - C - F

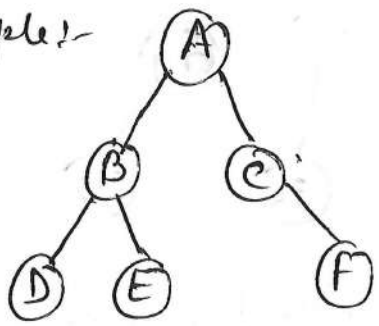
Preorder Sequence → 85, 40, 23, 36, 64, 72, 95.

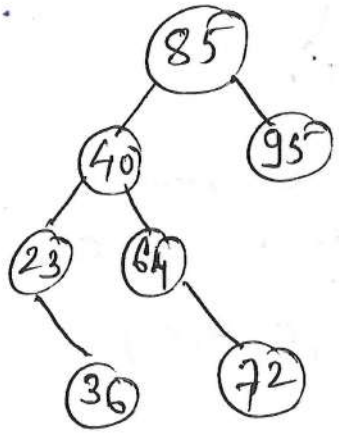
3) postorder Traversal :- In postorder sequence node are visited in following way :-

Left SubTree → Right SubTree → Root

Algorithm :- 1) Traverse the Left Sub Tree
2) Traverse the Right Sub tree.
3) visit the root.

Example :-





postorder Sequence → D - E - B - F - C - A

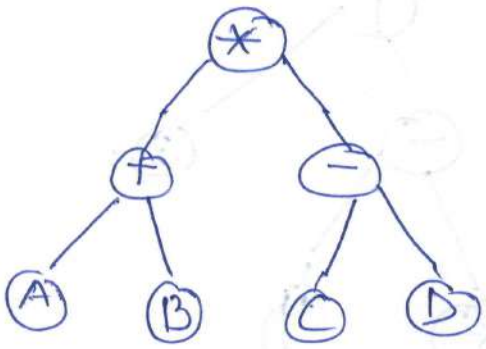
postorder Sequence → 36, 23, 72, 64, 40, 95, 85.

Expression Tree :- Expression Tree is a binary tree used to represent Expression.
Expression tree can be used to represent algebraic & boolean expression.

Structure :-
- Leaf node contains operands (numbers or variables)
- Internal nodes contains operators (+, -, *, /...)

Example :-
1) Construct expression Tree for given expression.
(A + B) * (C - D)

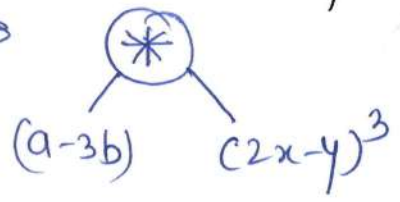
→



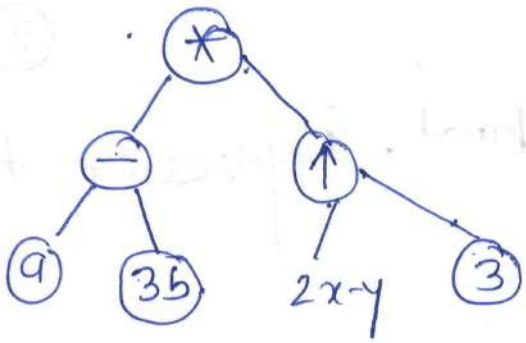
Expression Tree

2) Construct Expression Tree for $(a-3b)(2x-y)^3$

→

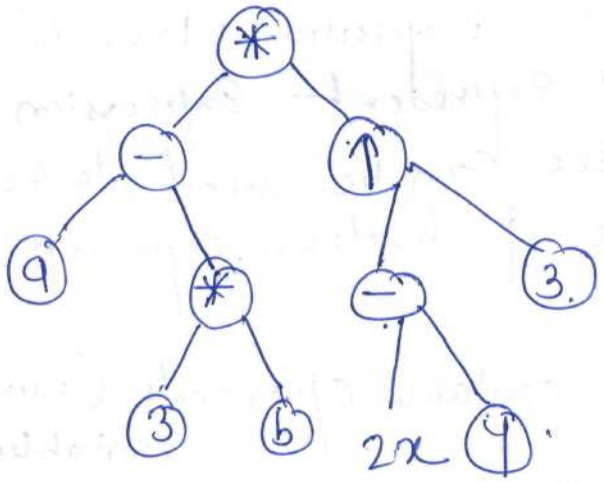


Step (1)

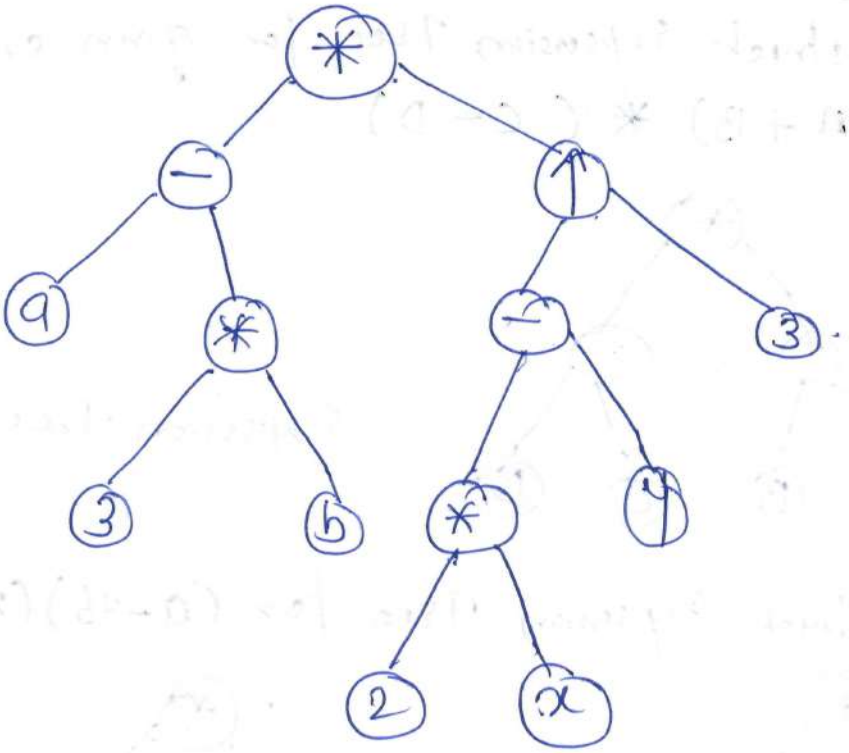


(2)

(3)



(4)



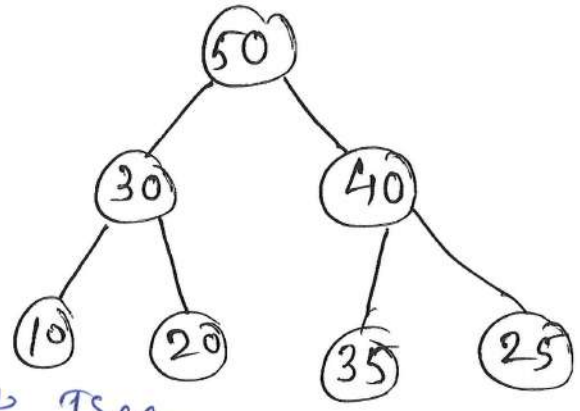
Final. Expression tree.

Heap : A Heap is a complete binary tree data structure that satisfies the heap property.

Types of Heap:-

1. Max Heap: In a Max-heap, the value of each child is less than or Equal to its parent. or : Every parent node is greater than or Equal to children or child node

Example:-

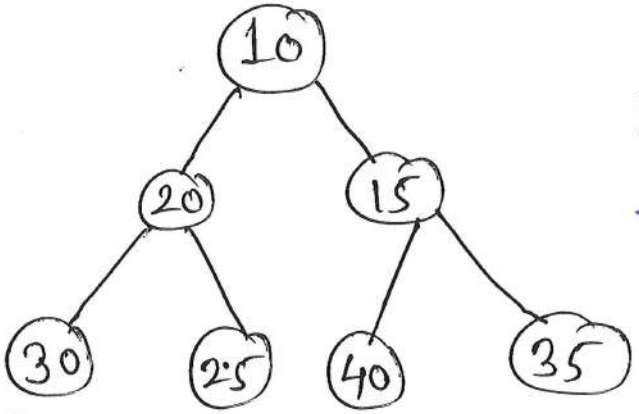


Max Heap Tree

Here,
50 > 30 & 40
30 > 10 & 20
40 > 35 & 25

2. Min Heap: In a Min-heap, the value of each child is greater than or Equal to its parent. or Every parent node is less than or Equal to children or child node.

Example:-



Here,
10 < 20 & 15
20 < 30 & 25
15 < 40 & 35

Min Heap Tree.

9/6/26

For (2 marks) **Question Bank.**

1. Describe the following term w.r.t. to tree : (SUM-2023)
   a) leaf node b) level of node c) Degree of tree.

2. Define the term binary tree and binary search tree. (SUM-2025)

3. Define heap data structure (SUM-2025).

4. Define the following terms with respect to tree:
   i) sibling. ii) Depth of tree. (WIN-2025)

5. Write algorithm for preorder traversal of binary tree. (WIN-2025).

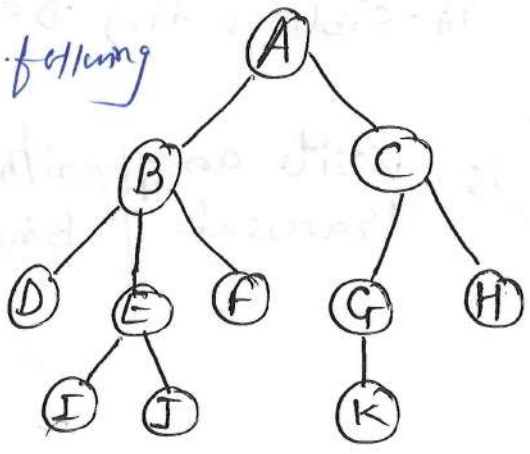
6. Define the following term w.r.t tree: (SUM-2024)
   i) In-degree ii) Out-degree

7. Define: i) General tree ii) Binary tree (SUM-2022)

For (4 or 6 marks).

8. Draw tree for given expression and find pre-order and post-order traversal. (SUM-2022)
   $(2b+5c)^2 (4d-6e)^5$.

9. From the given tree, complete following answers: i) Degree of tree
   ii) Degree of node B iii) Level of node H.
   iv) indegree of node C v) outdegree of node B vi) Height of tree.
   (SUM-2023)



10. Draw an expression tree for the following expression:
$(a+3b-7c)^3 * (6d-8e)^7$. (WIN-2023)

11. Draw tree for given expression :-
$(a-5b)^2 * (3x-7y)^3$. (WIN-2025)

12. From the given tree, find following in fig No.1. (WIN-2025).

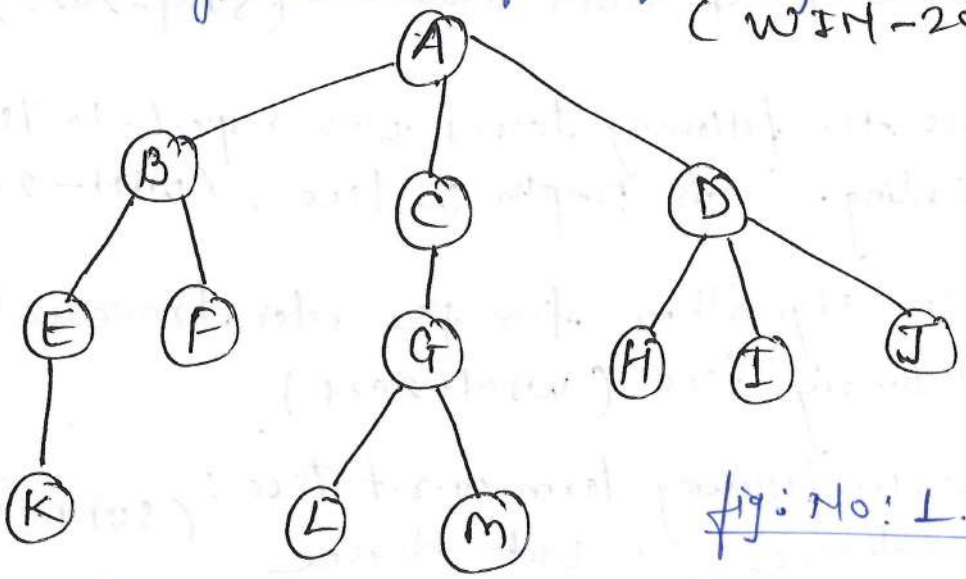


fig: No: 1.

i) Level of node M . ii) In-degree of node E
iii) Out-degree of node D . iv) Height of tree.

13. Explain Heap with example (WIN-2025).

14. Describe following terms with respect to tree data structure i) Leaf node of a tree (SUM-2024)
ii) Siblings iii) Degree of tree. iv) Depth of tree.

15. Write an algorithm for inorder and post-order traversal of Binary tree. (SUM-2025)

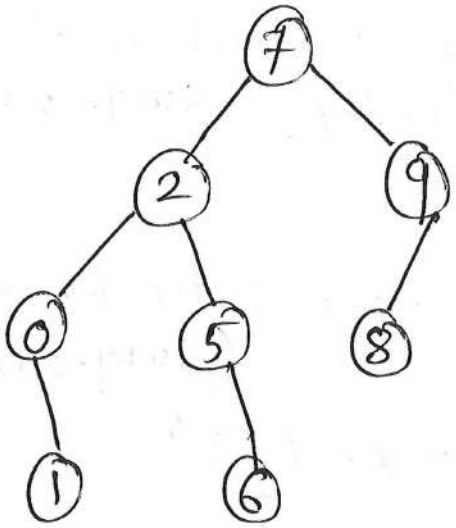
16. Construct a Binary search tree for the given numbers:
50, 33, 44, 22, 77, 35, 60, 40.
Prepare each step of construction of BST diagrammatically. (SUM-2025)

17. Draw the expression tree of the following expressions (SUM-2025)
i) $(2a+5b)^3 * (x-7y)^4$.
ii) $(a-3b) * (2x-y)^3$

18. Draw the tree structure for the following expression. (WIN-2024)
$(3A+7B) - [(6D-4E) \wedge 6C]$.

19. Construct a Binary search tree for following elements: (WIN-2024)
10, 3, 15, 22, 6, 45, 65, 23, 78, 34, 5.
Show each step of construction of BSP.

20. For given Binary tree write in-order, pre-order and post-order traversal.
(WIN-2024)



Ans -

9/6/26.