



SNS COLLEGE OF ENGINEERING

Kurumbapalayam (Po), Coimbatore - 641 107

An Autonomous Institution

Accredited by NBA - AICTE and Accredited by NAAC - UGC with 'A' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING-IOT Including CS&BCT

COURSE NAME : 19CS307 - DATA STRUCTURES

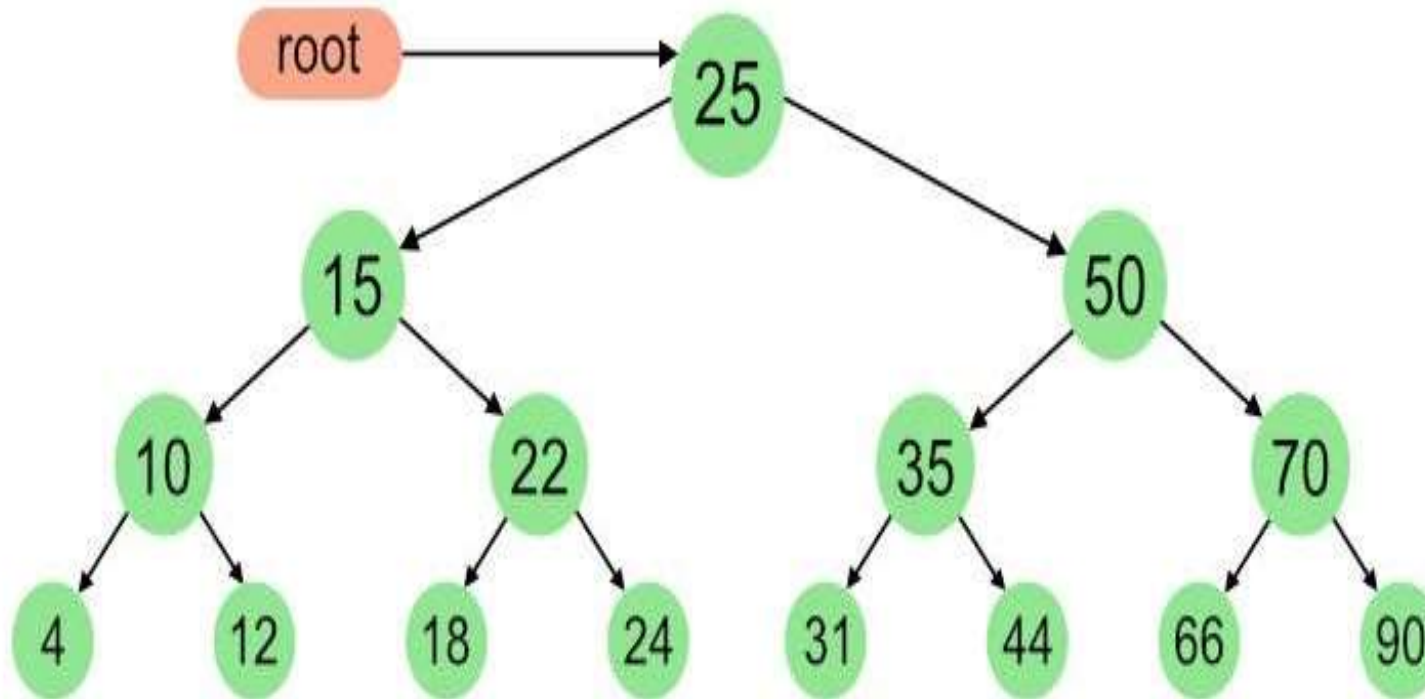
II YEAR / III SEMESTER

Unit III- NON LINEAR DATA STRUCTURES - Tree

Topic 2 : Tree Traversals



Problem





Tree Traversals



Unlike linear data structures (Array, Linked List, Queues, Stacks, etc) which have only one logical way to traverse them, trees can be traversed in different ways.

(Or)

Tree traversal (also known as **tree search** and **walking the tree**) is a form of [graph traversal](#) and refers to the process of visiting (e.g. retrieving, updating, or deleting) each node in a [tree data structure](#), exactly once. Such traversals are classified by the order in which the nodes are visited.



Tree Traversals



A binary tree node has data,
pointer to left child

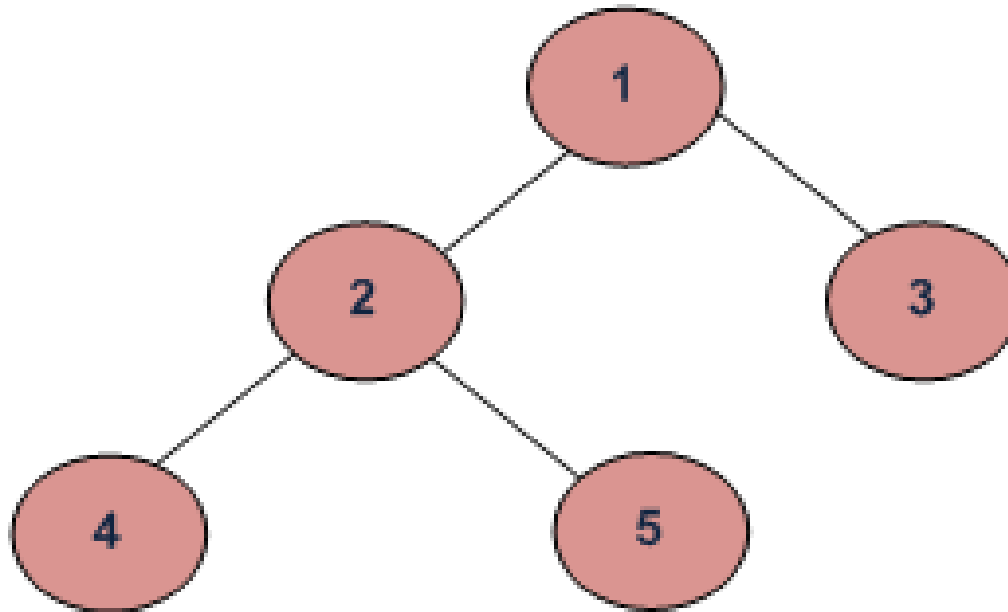
and a pointer to right child */

```
struct node {  
    int data;  
    struct node* left;  
    struct node* right;  
};
```

```
struct node* newNode(int data)  
{  
    struct node* node  
        = (struct  
node*)malloc(sizeof(struct  
node));  
    node->data = data;  
    node->left = NULL;  
    node->right = NULL;  
  
    return (node);  
}
```



Tree Traversals -Cont..



(a) Inorder (Left, Root, Right) : 4 2 5 1 3

(b) Preorder (Root, Left, Right) : 1 2 4 5 3

(c) Postorder (Left, Right, Root) : 4 5 2 3 1

Breadth First or Level Order Traversal : 1 2 3 4 5



Inorder Traversal



- Algorithm Inorder(tree)
- 1. Traverse the left subtree, i.e., call Inorder(left-subtree)
- 2. Visit the root.
- 3. Traverse the right subtree, i.e., call Inorder(right-subtree)

```
void printInorder(struct node*  
node)  
{  
    if (node == NULL)  
        return;  
  
    /* first recur on left child */  
    printInorder(node->left);  
  
    /* then print the data of node  
*/  
    printf("%d ", node->data);  
  
    /* now recur on right child */  
    printInorder(node->right);  
}
```

Inorder Traversal -Cont..

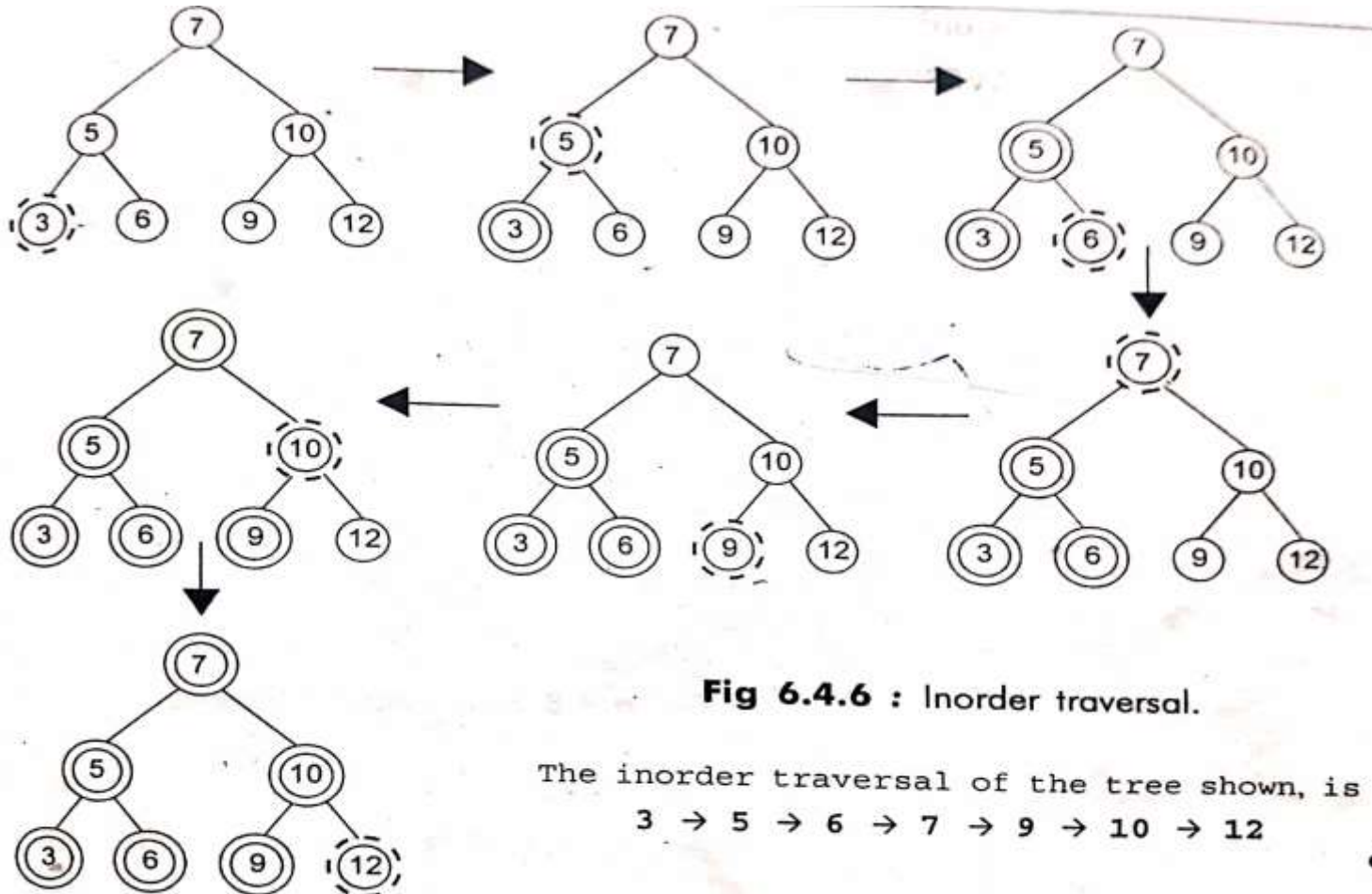


Fig 6.4.6 : Inorder traversal.

The inorder traversal of the tree shown, is
 3 → 5 → 6 → 7 → 9 → 10 → 12



• Preorder Traversal

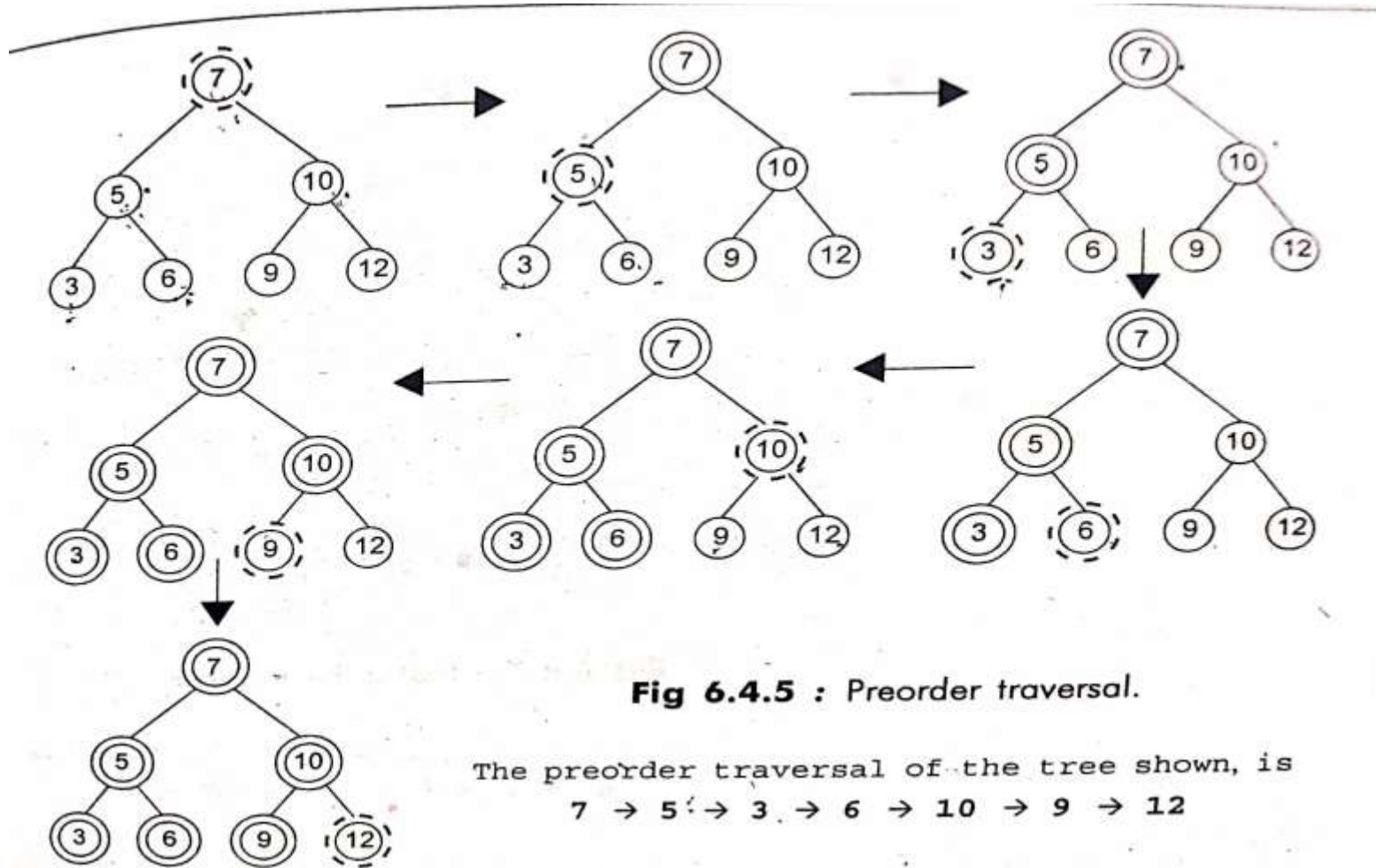
- Algorithm Preorder(tree)
- 1. Visit the root.
- 2. Traverse the left subtree, i.e. call Preorder(left-subtree)
- 3. Traverse the right subtree, i.e. call Preorder(right-subtree)

```
void printPreorder(struct node* node)
{
    if (node == NULL)
        return;

    /* first print data of node */
    printf("%d ", node->data);

    /* then recur on left subtree */
    printPreorder(node->left);

    /* now recur on right subtree */
    printPreorder(node->right);
}
```



Postorder Traversal

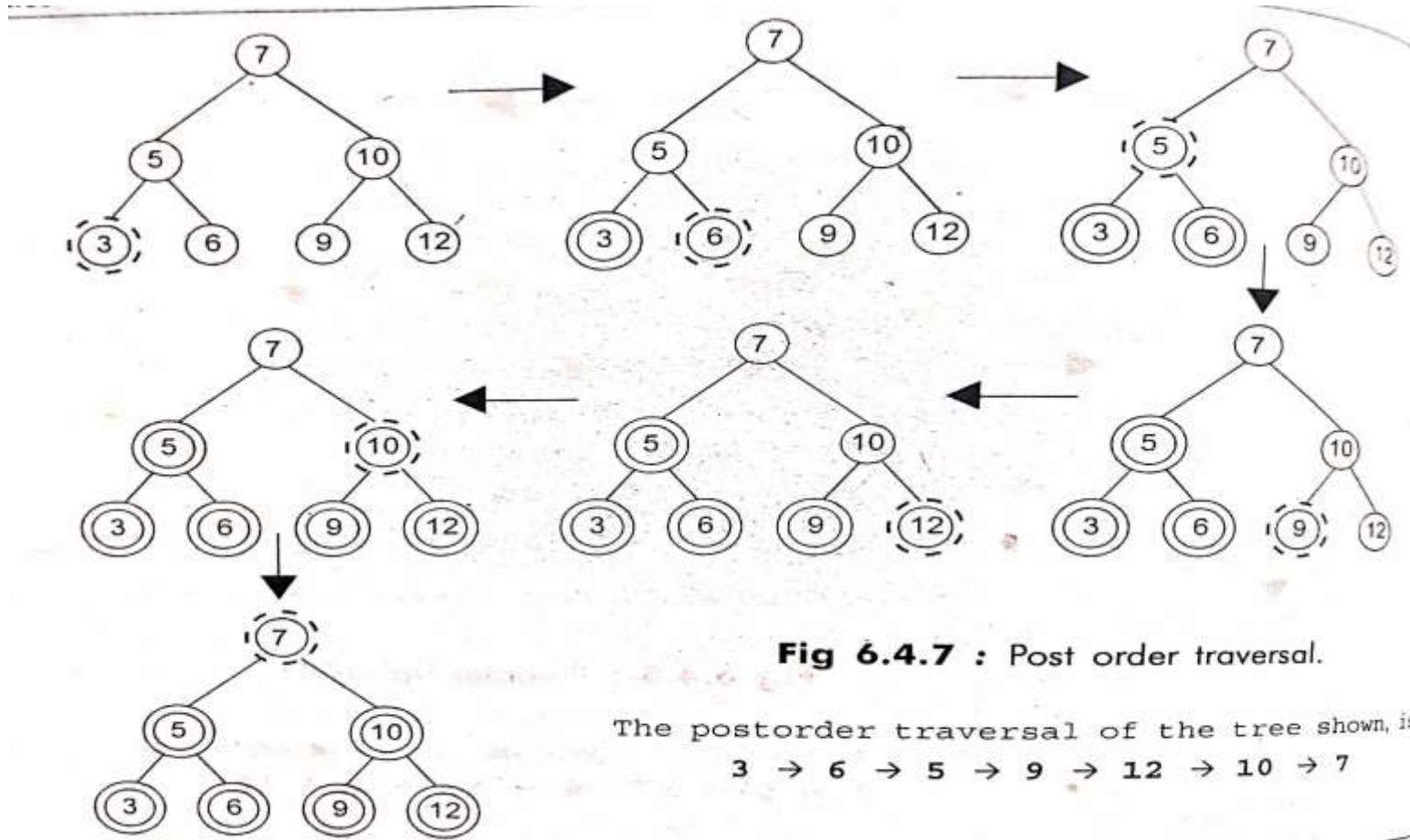


Algorithm Postorder(tree)

1. Traverse the left subtree, i.e., call Postorder(left-subtree)
2. Traverse the right subtree, i.e., call Postorder(right-subtree)
3. Visit the root.

```
void printPostorder(struct node*  
node)  
{  
    if (node == NULL)  
        return;  
  
    // first recur on left subtree  
    printPostorder(node->left);  
  
    // then recur on right subtree  
    printPostorder(node->right);  
  
    // now deal with the node  
    printf("%d ", node->data);  
}
```

Postorder Traversal -Cont..





Activity



MCQ



1. Which of the following graph traversals closely imitates level order traversal of a binary tree?

- a) Depth First Search
- b) Breadth First Search
- c) Depth & Breadth First Search
- d) Binary Search

2. From the following code identify the which traversal of a binary tree is this _____

```
//if node has left child order(node.left)
//if node has right child order(node.right)
visit(node)
```

- a) Inorder traversal
- b) preorder traversal
- c) postorder traversal
- d) Euler tour traversal



Advantages



Advantages:

- BST is fast in insertion and deletion etc when balanced.
- Very efficient and its code is easier than link lists.



Disadvantages



- Shape of the tree depends upon order of insertion and it can be degenerated.
- Searching takes long time.



Assessment 1



1. List out the advantages of Traversal Tree

- a) _____
- b) _____
- c) _____
- d) _____

2. Identify the disadvantages of Traversal Tree

- a) _____
- b) _____
- c) _____
- d) _____





REFERENCES



1. M. A. Weiss, “Data Structures and Algorithm Analysis in C”, Pearson Education, 8th Edition, 2007. [Unit I, II, III, IV,V]
2. A. V. Aho, J. E. Hopcroft and J. D. Ullman, “Data Structures and Algorithms”, Pearson Education, 2nd Edition, 2007 [Unit IV].
3. A.M.Tenenbaum, Y. Langsam and M. J. Augenstein, “Data Structures using C”, Pearson Education, 1st Edition, 2003.(UNIT I,II,V)
- 4.<https://www.geeksforgeeks.org/queue-set-1introduction-and-array-implementation/>

THANK YOU