Tree

~~What are the uses of Dep~~

1) Write an algorithm to compute the height of a binary tree. 2006

If tree is empty then return 0
else

(a) Get the max depth of left subtree recursively
i.e. → call maxDepth(tree → left-subtree)

(b) Get the max depth of right subtree recursively
i.e. → call maxDepth(tree → right-subtree)

(c) Get the max of max depths of left and right subtrees and add 1 to it for the current node.
max_depth = max(max depth of left subtree, max depth of right subtree) + 1

(d) Return max_depth

function in C

```
int maxDepth(node* node)
{
  if(node == NULL)
      return 0;
  else
  {
    int lDepth = maxDepth(node->left);
    int rDepth = maxDepth(node-> right);
    if (l_Depth > r_Depth)
        return (lDepth + 1);
    else
        return (r_Depth+1);
  }
}
```

2) Write an algorithm to count the number of nodes in a binary tree. 2006

```
int countnodes (struct node * root)
{
  if (root != NULL)
  {
    countnodes (root->left);
    count ++;
    countnodes(root->right);
  }
  return count;
}
```

3) Prove that the maximum number of nodes on level i of a binary tree is $2^i - 1$, $i \geq 1$ 2006

Answer given in notes.

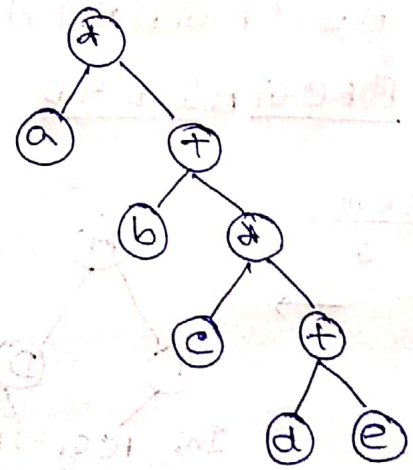
4) Prove that the maximum number of nodes in a binary tree of depth k is $2^k - 1$; $k \geq 1$. 2006

Same as 4)

5) Draw the tree representation for the prefix expression: *a + b*c + de. 2006

Infix → a * b + (c * (d + e))

see notes



6) What is binary search tree? 2007, 2011, 2013

7) Distinguish between DFS and BFS. Indicate their time complexities. 2008, 2011

8) What are B-tree and B+tree? Give the difference between them. 2009, 2012

9) Insert the following keys into a Btree of order 3: a, f, b, k, h, m. Do it by yourself

10) Show the stages in growth of 4 order B-tree when the following keys are inserted in the given order: 2013

74 72 19 87 51 10 35 18 39 60 76 58 1995

Do it

11) Define the following terms:
   a) complete Binary tree
   b) Degree of a tree
   c) weighted graph

given in notes

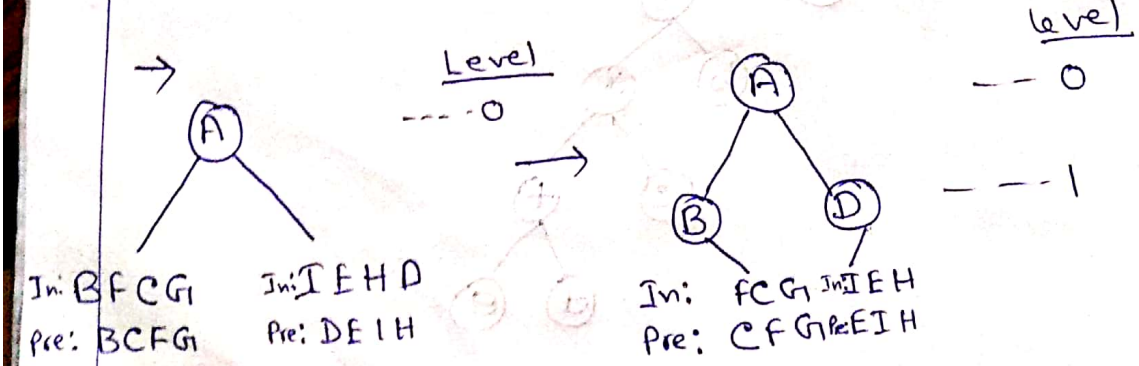
12) Make a BST from the given traversals-

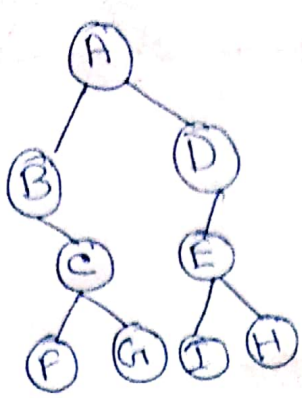
Preorder: A B C F G D E I H

Inorder: B F C G A I E H D

NLR

LNR





13) Write an algorithm/c-function for preorder traversal of a binary tree. 2015 already in notes

14) Consider the following sequence of binary tree traversals.

Inorder: Q B K C F A G P E D H R

Preorder: G B Q A C K F P D E R H

Do it by yourself.

15) Define the following terminologies:
1) Node 2) Root 3) Siblings 4) level
5) leaf node. 2015

16) What is a graph? When a graph will be a tree?

17) How a binary tree is different from a binary search tree? in notes

18) What do you mean by "threading in a binary tree? When threaded binary tree is useful? in notes 2004, 2015

19) Construct the binary search tree if the elements are in order:
60, 75, 25, 66, 50, 55, 45, 40, 35, 57, 30
Delete the following nodes in order and show each step:
i. node with 55
ii. node with 66
iii. node with 50

Do it by yourself about deletion if you need help see notes or contact me

20) Consider the following sequence of a binary tree traversals.

Inorder: BCEDFAGH

Preorder: ABCDEFGH

Hence, construct the binary tree.

21) Write an algorithm for post order traversal of a binary tree. 2004

22) The order of nodes of a binary tree in Pre-order traversals are as follows:

Preorder: A B C D F H J M K E G I L N

Inorder: A D J M H K F C I N L G E B

Draw the corresponding binary tree.

23) a) Construct a binary search tree if the elements are in the order: 2006

56, 78, 34, 90, 89, 91, 39, 36

b) Delete the following nodes in order and show each step.

i) node with 36

ii) node with 39

iii) node with 56.

24) Draw all possible binary search trees containing the 4 elements. 2006

25) Write a non recursive function to traverse a binary tree using inorder traversal. 2007, 2012

26) Write an algorithm to implement binary search tree. Also write the algorithm to delete an element from a binary search tree.

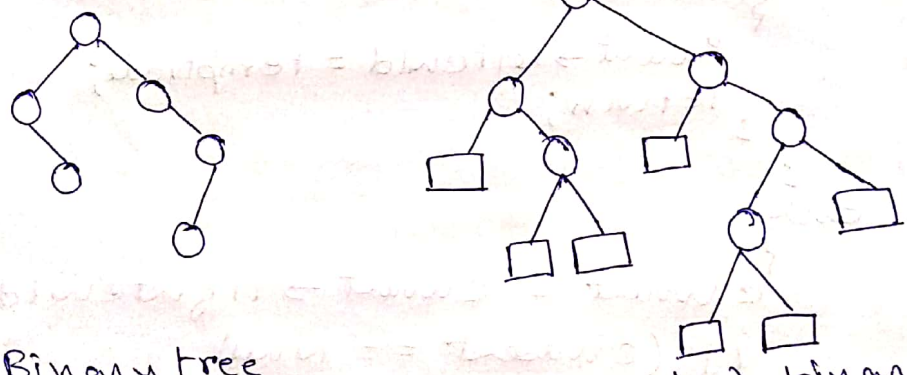
```
void insert (int data)
{
struct node *tempNode = (struct node*) malloc
                        (sizeof(struct node));
struct node *current;
struct node * Parent;
tempNode -> data = data;
tempNode -> leftchild = NULL;
tempNode -> rightchild = NULL;
if(root == NULL)
{
root = tempNode;
}
else
{
current = root;
Parent = NULL;
while(1)
{
parent = current;
if (data < Parent -> data)
{
current = current -> leftchild;
if (current == NULL)
{
Parent -> leftchild = tempNode;
return;
}
}
else
{
current = current -> rightchild;
if (current == NULL)
{parent -> rightchild = tempNode;
return;
}
}
}
}
}
```

27) Write a function to delete any node from a binary search tree.

28) What is Extended Binary tree? 2012

→

If in a binary tree, each empty subtree (NULL link) is replaced by a special node then the resulting tree is extended binary tree or 2-tree. So we can convert a binary tree to an extended binary tree by adding special nodes to leaf nodes and nodes that have only one child. The special nodes added to the tree are called external nodes and the original nodes of the tree are internal nodes. The following figure shows a binary tree and the corresponding extended binary tree.



Binary tree

Extended binary tree

In the figure, external nodes are shown by squares and internal nodes by circles. We can see that all the external nodes are leaf nodes while the internal nodes are non leaf nodes.

The extended binary tree is a strictly binary tree i.e each node has either 0 or 2 children

29) Construct the binary search tree if the elements are in the order.
60, 70, 30, 20, 55, 90, 95, 80, 55, 35, 45, 40, 50
Insert the following nodes in order and show each step.
i) node with 25
ii) node with 65

30) What are the differences between general tree and a binary tree?
Define General tree. Write an algorithm to convert a General tree into a binary tree.

→ Definition of General tree: General trees are those in which the number of subtrees for any node is not required to be 0, 1 or 2.

The process of converting the general tree to a binary tree is as follows:

Step 1 → Use the root of the general tree as the root of the binary tree.

Step 2 → Determine the first child of the root. This is the leftmost node in the general tree at the next level.

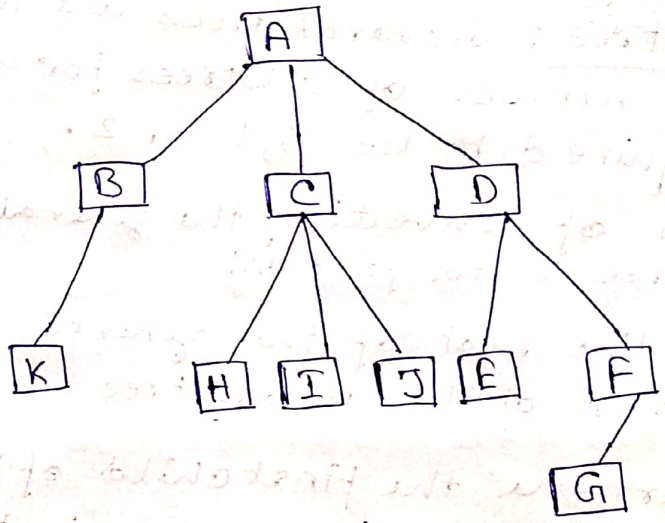
Step 3 → Insert this node. The child reference of the parent node refers to this node.

Step 4 → Continue finding the first child of each parent node and insert it below the parent node with the child reference of the parent to this node.

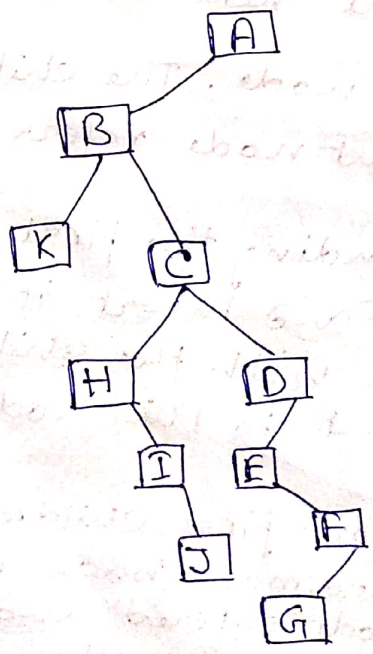
Step 5 → When no more first children exist in the path just used, move back to the parent of the last node entered and ~~present~~ repeat the above process. In other words, determin

Steps → complete the tree for all nodes. In order to locate where the node fits you must search for the first child at that level and then follow the sibling references to a nil where the next sibling can be inserted. The children of any sibling node can be inserted by locating the parent and then inserting the first child. Then the above process is repeated.

Given the following general tree



The following is the binary version:



Differences between General tree and a binary tree

General tree	binary tree
A general tree is a data structure in that each node can have infinite number of children.	A binary tree is a data structure in that each node has at most two nodes, left and right.
A general tree can't be empty.	A binary tree can be empty.
There is no limit on the degree of node in a general tree.	nodes in a binary tree cannot have more than degree 2.
Subtree of general tree are not ordered.	Subtree of binary tree are ordered.

31) Make Binary Search tree from the following numbers and do inorder, preorder and Postorder traversals: 2014

56, 34, 45, 37, 48, 87, 63, 75, 59, 94, 67

32) Write a recursive algorithm for Preorder and Postorder traversals of a binary tree. Do it yourself note is given

33) Short note on the following:

a) B-tree
b) Linearization of a binary tree.
c) B^+ - tree
d) In-order, Pre-order, Post order
e) Threaded binary tree
f) Tree traversal Algorithm (Binary tree Traversal → In → Pre → Post)