Q.21 A full 3-ary tree has 4 non-leaf nodes, how many leaf nodes does it have?

Q.22 Consider the following code.

```
find (*T1, *T2)
{    while(T1 ! = NULL && T2 ! = NULL)
     {    if ((T1 → data) = = (T2 → data))
          {    find (T1 → left, T2 → left);
               find (T1 → right, T2 → right);
          }
          else {printf ("GATE"); exit ( );}
     }
}
```

Find when "GATE" will not be printed by the above program? [Assume two binary trees are passed into the above function].

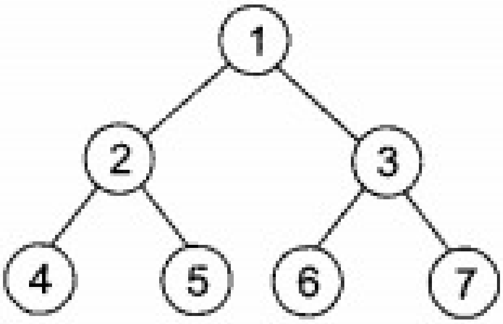
(a) If the two trees are same

(b) If the two trees are different

(c) If the two trees have same levels

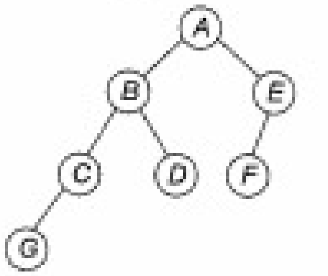
(d) If the two trees have different levels

Q.23 Consider the following tree : If the post order traversal gives $ab - cd * +$ then the label of the nodes 1, 2, 3, ... will



(a) +, −, *, *a*, *b*, *c*, *d*
(b) *a*, −, *b*, +, *c*, *, *d*
(c) *a*, *b*, *c*, *d*, −, *, +
(d) −, *a*, *b*, +, *, *c*, *d*

Q.24 A balanced tree is given below



How many nodes will become unbalanced when a node is inserted as a child of node *G*?

(a) 2 (b) 3
(c) 4 (d) 5

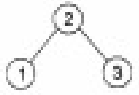
Q.25 In delete operation of binary search tree, we need inorder successor (or predecessor) of a node when a node to be deleted where it has both left and right child. Which of the following is true about inorder successor needed in delete operation?

(a) Inorder successor is always either leaf node or a node with empty right child.
(b) Inorder successor maybe an ancestor of the node.
(c) Inorder successor is always a leaf node.
(d) Inorder successor is always either a leaf node or a node with empty left child.

Q.26 Consider the following program.

```
void find (struct Node *node)
{   struct Node *ptr, *q;
    if (node == NULL) return;
    find (node → left);
    find (node → right);
    ptr = node → left;
    node → left = newNode (node → data);
    node → left → left = ptr;
}
```

If the root of following tree is passed to the above function, what is the level order traversal of output tree produced by above function? (newNode is a function which creates new node)



(a) 2 2 3 3 1 1 (b) 2 2 3 1 3 1

(c) 2 3 2 3 1 1 (d) 2 3 2 3 2 1

Q.27 The minimum size that an array may require to store a binary tree with '*n*' nodes is ______.

(a) $2^{\lceil \log_2(n+1) \rceil} - 1$ (b) $2^n - 1$

(c) $2^n - n + 1$ (d) $n + 1$

Q.28 Consider the following Binary search tree with the following traversals.

Inorder: *A, M, N, O, P, Q, R, S, T, X, Z*

Preorder: *Q, P, A, M, N, O, X, S, R, T, Z*

Find the number of elements in the 4th, 5th and 6th level respectively. Assume root is at level 1.

(a) 3, 2, 2 (b) 2, 2, 2

(c) 3, 1, 1 (d) 3, 2, 1

Q.29 The depth of a complete binary tree with '*n*' nodes is (log is to the base two)

(a) $\log(n + 1) - 1$ (b) $\log(n)$

(c) $\log(n - 1) + 1$ (d) $\log(n) + 1$

Q.30 The number of possible binary search trees with 3 nodes is

(a) 12 (b) 13

(c) 5 (d) 15

Q.31 A binary tree in which every non-leaf node has non-empty left and right subtrees is called a strictly binary tree. Such a tree with 10 leaf nodes

(a) cannot have more than 19 nodes

(b) has exactly 19 nodes

(c) has exactly 17 nodes

(d) cannot have more than 17 nodes

Q.32 Consider the following statements:

1. Infix, Prefix and Postfix notations for expressing sum of *A* and *B* are *A*+*B*, +*AB*, and *AB*+, respectively.
2. AVL tree is a binary tree in which the difference in heights between the left and the right sub tree is not more than one for every node.
3. Stack data structure is used to save and retrieve information in reverse order.
4. Queue data structure is known as LIFO.

Which of the statements given above are correct?

(a) 1, 2 and 3 (b) 2, 3 and 4

(c) 1, 3 and 4 (d) 1, 2 and 4

Q.33 A program takes an input a binary tree with '*n*' leaf nodes and computes the value of function $g(x)$ for each node '*I*'. If the cost of computing $G(x)$ = (Number of leaf nodes in left subtree of *I*- minimum number of node in right subtree of *I*), then the worst case time complexity of the program is

(a) $O(n)$ (b) $O(n^2)$

(c) $O(n \log n)$ (d) $O(\log n)$

Q.34 The key 14, 4, 6, 16, 32, 50 in the order are inserted into an initially empty AVL tree. Find total number of rotations to make AVL with the given keys. Assume "single rotation = 1 rotation" and "double rotation = 1 rotation".

Q.35 Let *T* be a binary search tree with 160 vertices. The smallest possible height of *T* is ______ (Assume root at height 0).

Q.36 Given pre-oder and post-oder traversal of binary search tree.

Pre-oder : 60, 37, 25, 21, 23, 39, 38, 50, 70, 65, 100, 85, 110.

Post-oder : 23, 21, 25, 38, 50, 39, 37, 65, 85, 110, 100, 70, 60

The maximum number of node presents at any level is ______.

■■■■

21 ⇒ Full 3-ary tee ~~4-non~~ leaf node , leaf-node => 5

9

→ leaf node = 9

Ans

22 ⇒

A

```
find (*T1 , *T2)
{ while (T1 != NULL && T2 != NULL)
   {  if ((T1->data) == (T2->data))
      { find (T1 -> left , T2 -> left);
        find (T1 -> right , T2 -> right);
      }
      else { printf("GATE"); exit(); }
   }
}
```

→ find when "GATE" not printed

→ If two trees are same

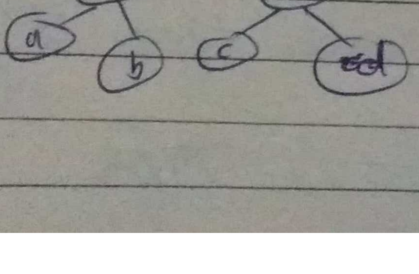
Ans

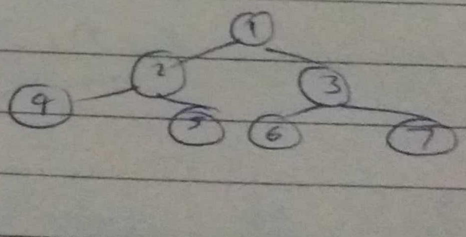
23 ⇒

A

past-order → ab- cd * + ·

level of node





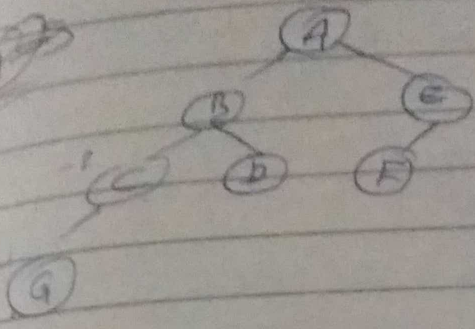
1, 2, 3, 4, 5, 6, 7

= +, - * a b c d

Ans

24

B



No of node unbalanced when a node insert in G as child

Node A, B, C → height = 2

unbalanced

No. of node = 3 Ans

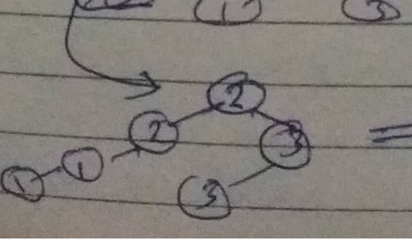
25 ⇒ BST delete node ~~as~~ with both child (inorder)

D

Inorder successor is always either leaf node or a node with empty left child. Ans

26 ⇒ If root pass in f^n, then level-order = ?

B



⇒ 2, 2, 3, 1, 3, 1 Ans

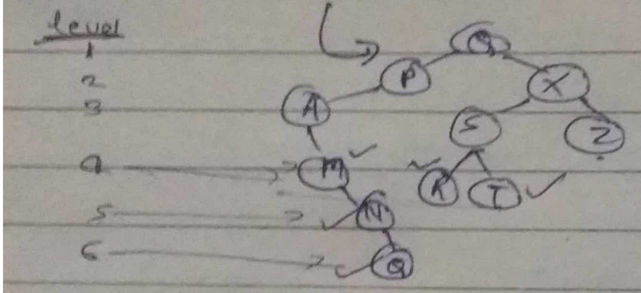
(27) → minimum size of array to store BT of n-node

(A) ↳ $2^{\lceil \log(n+1) \rceil - 1}$ Ans

(28) → Inorder → A, M, N, O, P, Q, R, S, T, X, Z

Preorder → Q, P, A, M, N, O, X, S, R, T, Z

(d) ↳ No of element at 4th, 5th, 6th, (start (1)),



⟹ 3, 1, 1 Ans

(29) → Depth of complete binary tree with n-node –

(A) ↳ $\log(n+1) - 1$ Ans

(30) → No. of possible BST with 3-node is →

(C) ↳ 5 Ans

(31) → strictly binary tree has 10 leaf node

(B) ↳ has exactly 19 nodes Ans

(32) → Current statements –

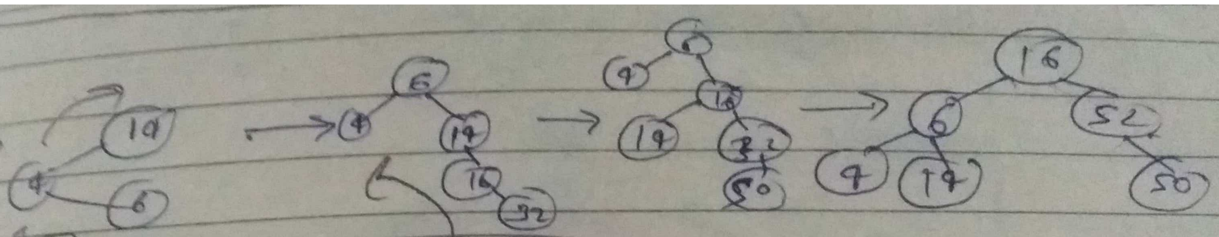
(A) ↳ 1, 2, ~~&~~ 3 Ans

(33) → I/P → BT with n- leaf node, worst case time-compl-

(A) ↳ O(n) Ans

34

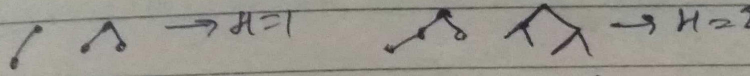
3



Total 3 rotation Ans

35

7



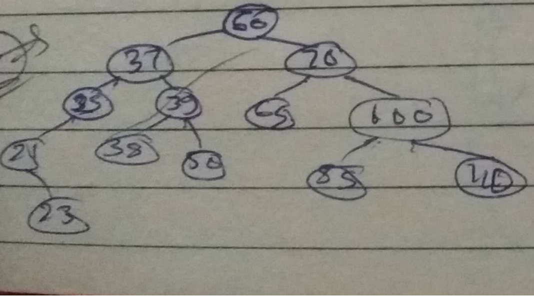
smallest height possible in (complete tree) = $H = \lceil \log(n+1) \rceil - 1$

$H = \lceil \log_2(160+1) \rceil - 1 = 8-1$

$H=7$ Ans

36

5



⟹ max node 5 at level 3

Ans