CS & IT ENGINEERING



Trees-3

DPP-03 Discussion Notes



By- Pankaj Sharma sir



TOPICS TO BE COVERED

01 Question

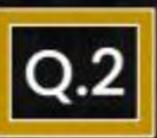
02 Discussion

The number of unlabelled binary trees possible with four nodes is



[NAT]

#unlabelled binasy trees with n nodes =
$$\frac{2n}{n+1}$$
 $n=u \Rightarrow \frac{8c_u}{5} = \frac{81}{5 \times 4141} = \frac{8 \times 7 \times 8 \times 9 \times 91}{9 \times 94 \times 94}$
 $\frac{3}{3}$



The number of labelled binary trees possible with the nodes-10, 30, 25, 40 is 336. [NAT]



labelled binary trees with n nodes =
$$\frac{3n}{n+1} \times n!$$
 $n = 4 \Rightarrow \left(\frac{8c_4}{4+1}\right) \times 4!$
 $= 14 \times 4! = 14 \times 24$
 $= 336$

Q.3

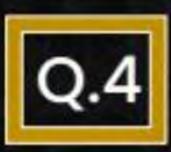
The number of binary search trees possible with the nodes-10, 30, 25, 40 is _____.



[NAT]

labelled BST with
$$n - \text{Reys} = \left(\frac{2n}{n+1}\right)$$

labelled BST with 10,30,75,40 =
$$\frac{8}{4+1} = \frac{14}{4+1}$$



The pre-order traversal of a binary search tree is given as- [MCQ] 7, 3, 2, 1, 5, 4, 6, 8, 10, 9, 11
The post-order traversal of the above binary tree is-





1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11

B.

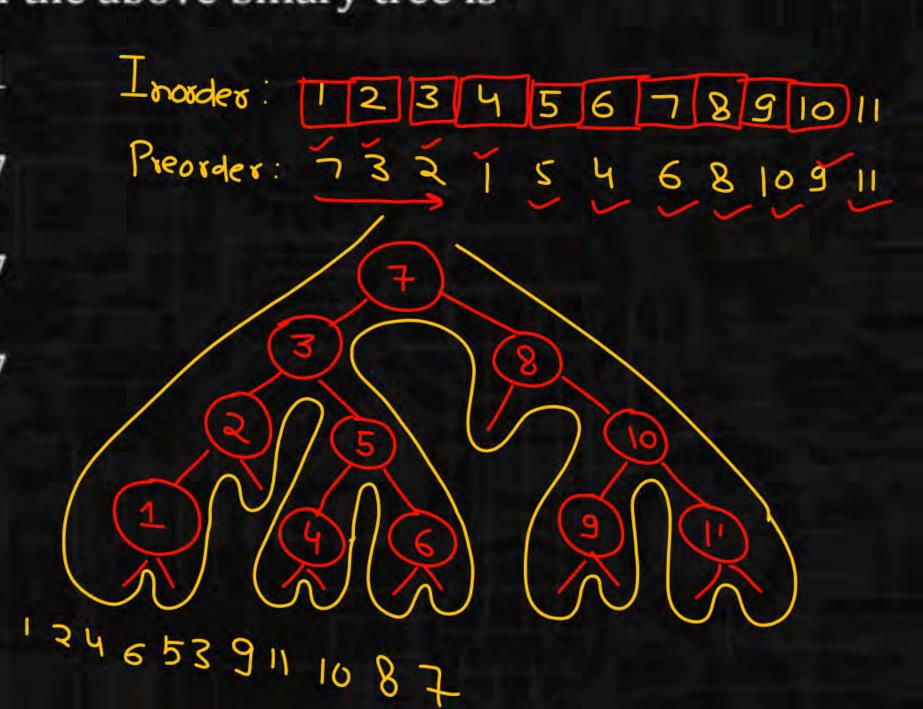
1, 2, 4, 6, 5, 3, 9, 11, 10, 8, 7

G.

1, 2, 4, 5, 6, 3, 9, 10, 11, 8, 7

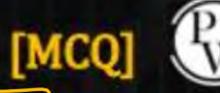
D.

11, 9, 10, 8, 6, 4, 5, 1, 2, 3, 7





Consider the following two statements:



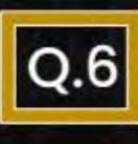
Statement P: The last elements in the pre-order and in-order traversal of a binary search tree are always same. Incorrect Statement Q: The last elements in the pre-order and in-order traversal of a binary tree are always same. Incorrect Which of the following tree is/are CORRECT?

- A. Both P and Q only
- B. Neither P nor Q
- c. Q only
- D. Ponly

Pre: 105

In: 5 10

5



Consider the following function:

12

struct treenode {
struct treenode *left;
int data;
struct treenode *right;
};

10

```
int func (struct treenode *t){
if(t==NULL) return 1;
else if(t->left==NULL && t->right==NULL)
```

return 1;

else if

 $((t\rightarrow left\rightarrow data < t->data) && (t\rightarrow right\rightarrow data > t->data))$

return func(t->left) && func(t->right);

else

return 0;

Assume t contains the address of the root node of a tree. The function-



Returns 1 if the given tree is a Binary Search Tree.



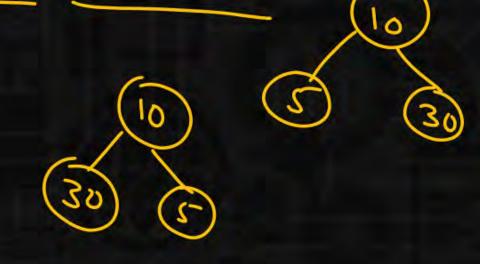
Returns 0 if the given tree is a complete binary tree.



Returns 0 if the given tree is a Binary Search Tree.



Returns 1 if the given tree is a complete binary tree.



NULI

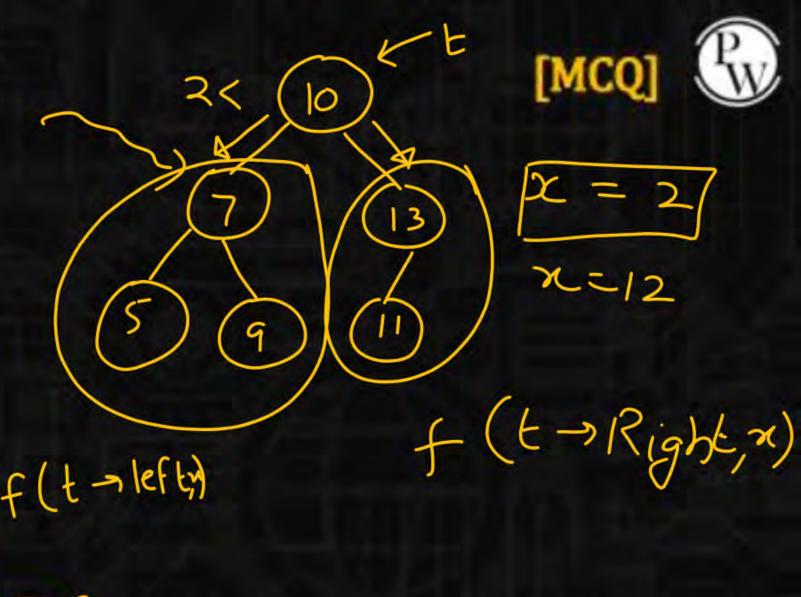
NULL [MCQ

NULL

NULL



Consider the following function: struct treenode{ struct treenode *left; int data; struct treenode *right; struct treenode * f(struct treenode *t, $int x){$ if(t==NULL) return NULL; elseif(x==t->data) return else if (x<t->data) return_ else return Assume t contains the address of the root node of a binary search tree. The function finds an element x in the BST and returns the address of the node if found. Which of the following is/are CORRECT?



a: NULL; b:

a: NULL; b: f(t->left, x); c: f(t->right, x)

B.

a: t; b: f(t->right, x); c: f(t->left, x)



a: NULL; b: f(t->right, x); c: f(t->left, x)



a: t; b: f(t->left, x); c: f(t->right, x)



