CS & IT ENGINEERING

Data Structures

Tree

Lecture No.- 03

By- Pankaj Sharma Sir

Topics to be Covered

Topic Tree Part-03

binary trees

Recap of Previous Lecture

Topic Tree Part-02

Tree traversal
- pre-order
- In-order
- Post-order

Topic : Tree

Pankaj sir PW

1. Unlabelled
2. labelled

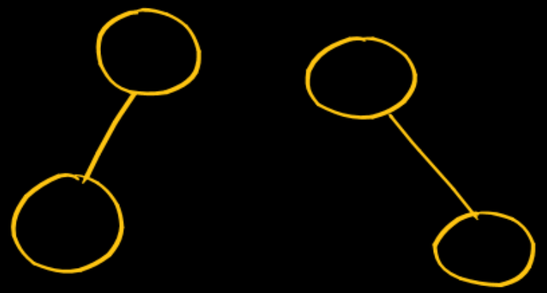
unlabelled binary trees with n nodes

(shape / structure / topology)

$n = 1$

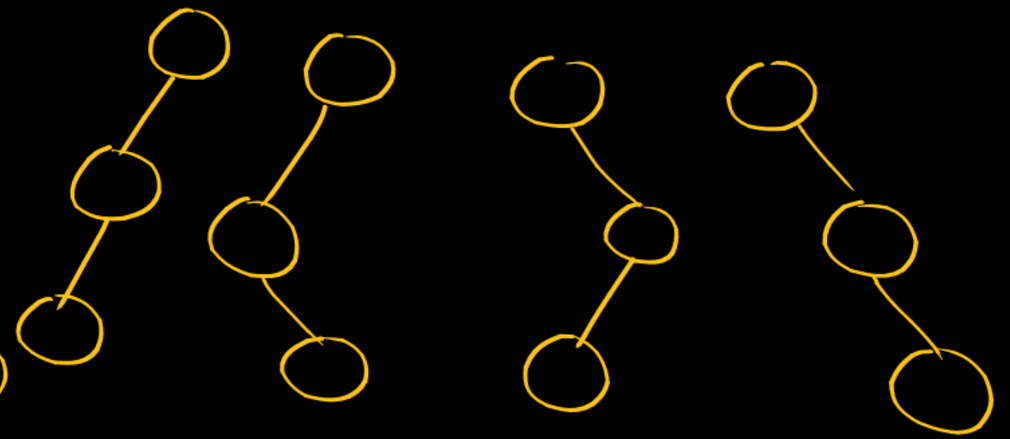
1 ✓

$n = 2$



2 ✓

$n = 3$



5

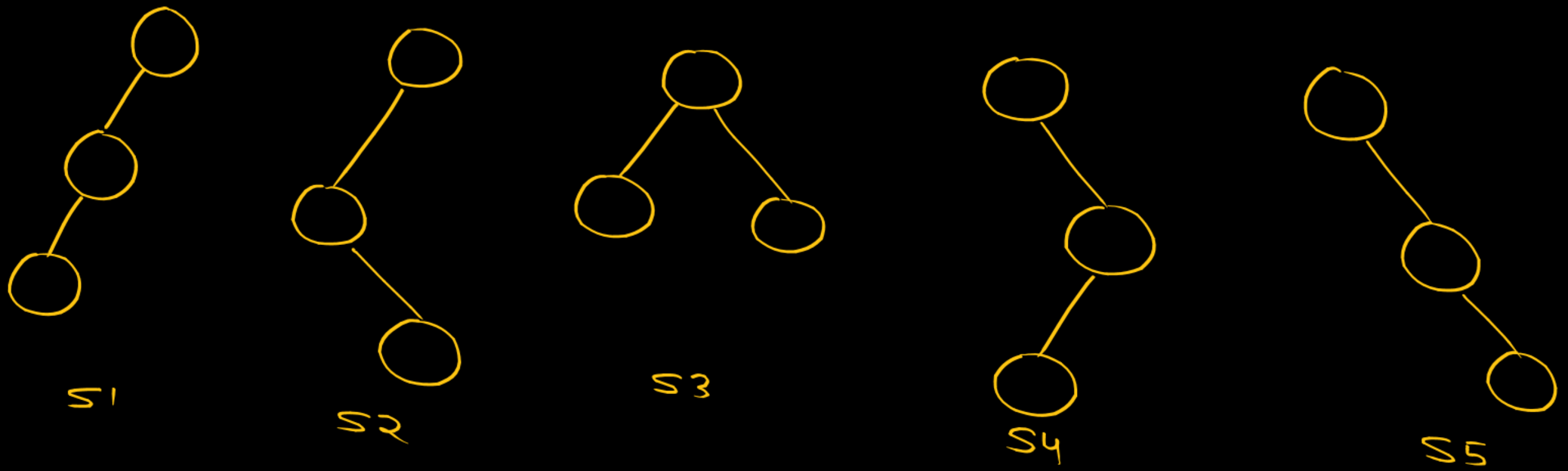
unlabelled binary tree with n nodes = $\frac{{}^{2n}C_n}{n+1}$

$n = 3$

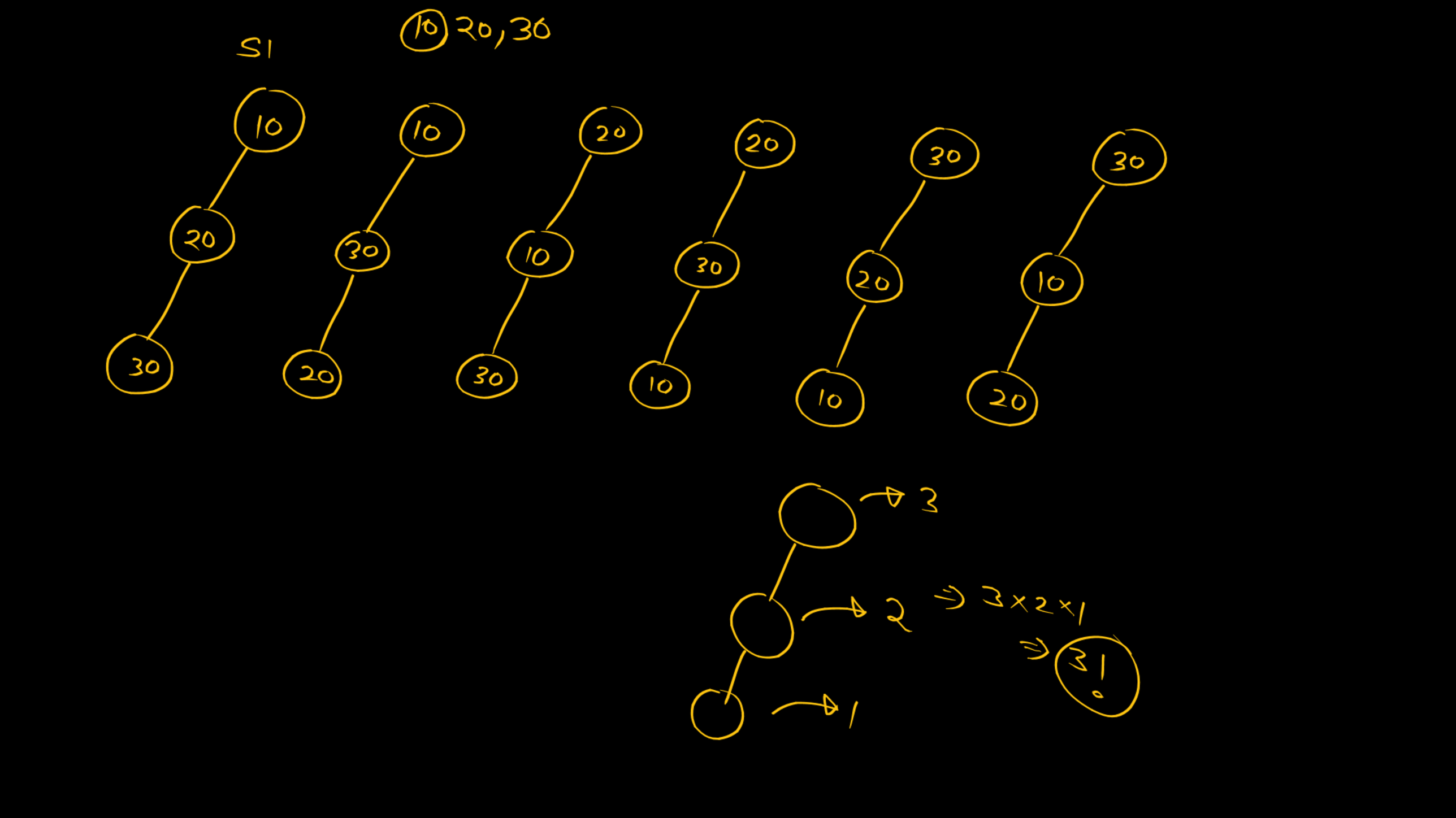
$$\frac{{}^{6}C_3}{3+1} = \frac{{}^{6}C_3}{4} = \frac{1}{4} \times \frac{6!}{3!3!} = \frac{1}{4} \times \frac{6 \times 5 \times 4 \times 3!}{3! \times 3!} = 5$$

With 3 nodes $\Rightarrow$ 5 binary tree structure/shape are possible.

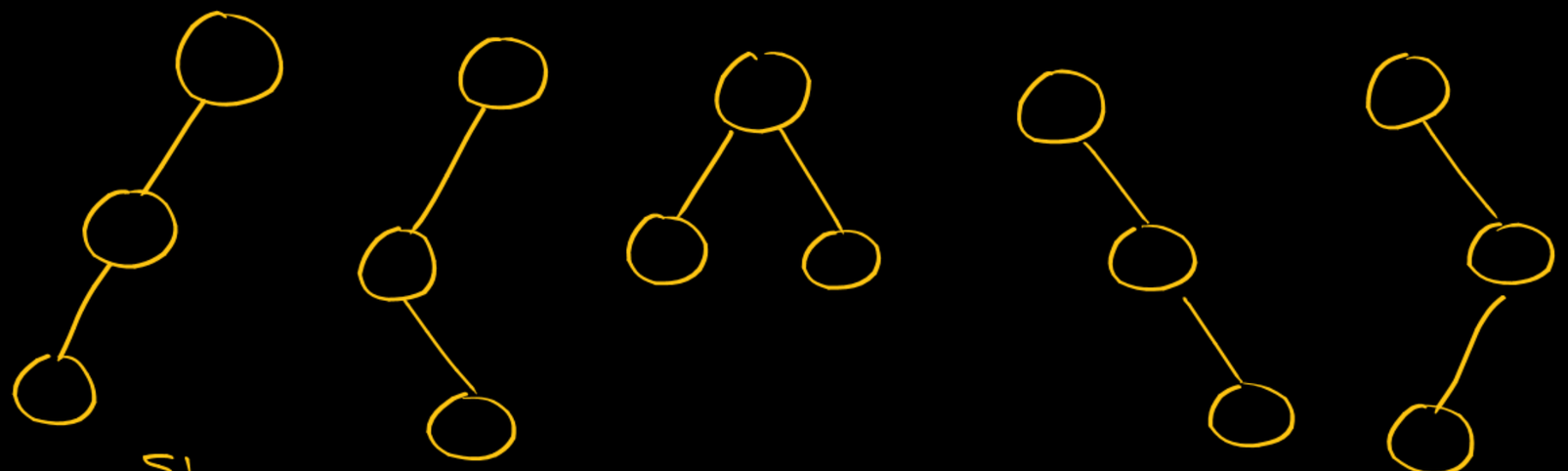
$n = 3 \rightarrow 5$ structure are possible



labelled binary tree with 3 distinct keys.


S1
10 20,30
10
20
30
10
30
20
20
10
30
30
20
10
10
20
30
10
20
3
2
3x2x1
3!
1

$n = 3$



S_1 S_2 S_3 S_4 S_5

$3!$ $3!$ $3!$ $3!$ $3!$

labelled binary trees with 3 distinct keys

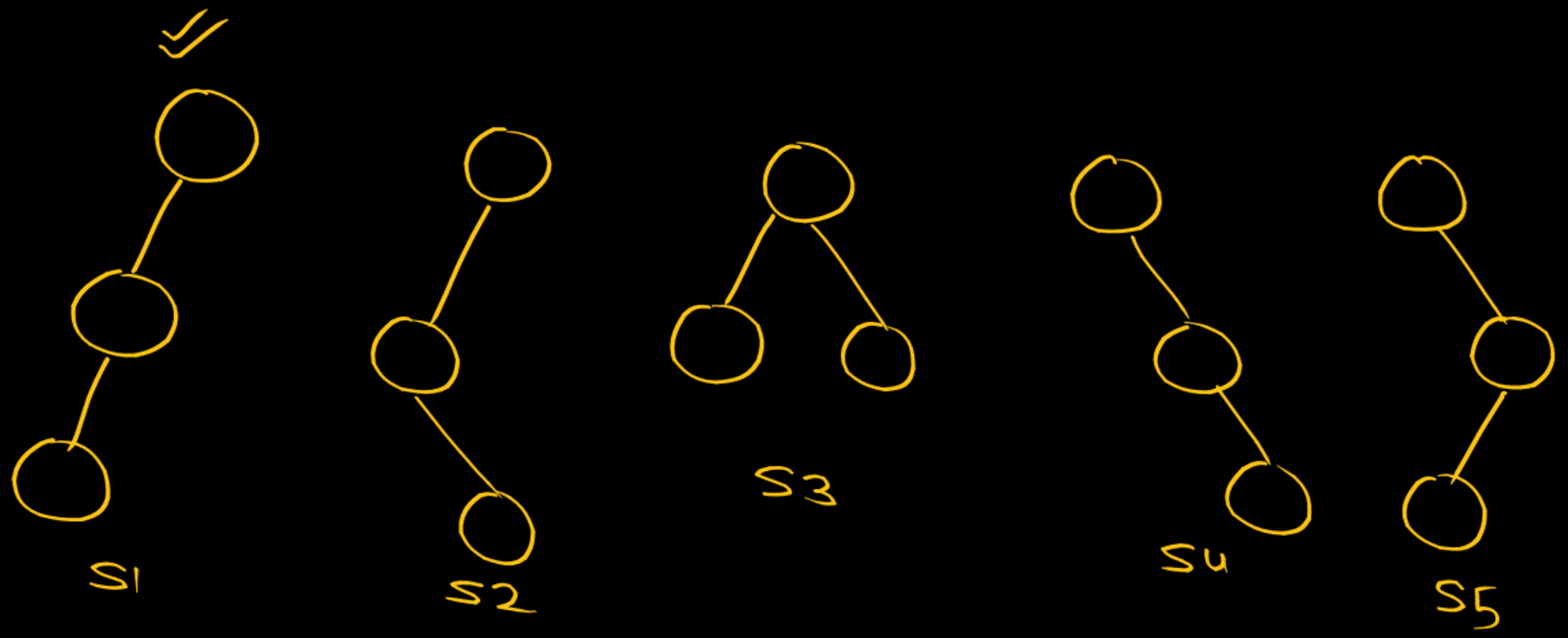
$= \left(\begin{array}{c} \text{\# of unlabelled binary} \\ \text{trees with 3 nodes} \end{array} \right) \times 3!$

$= 5 \times 6$

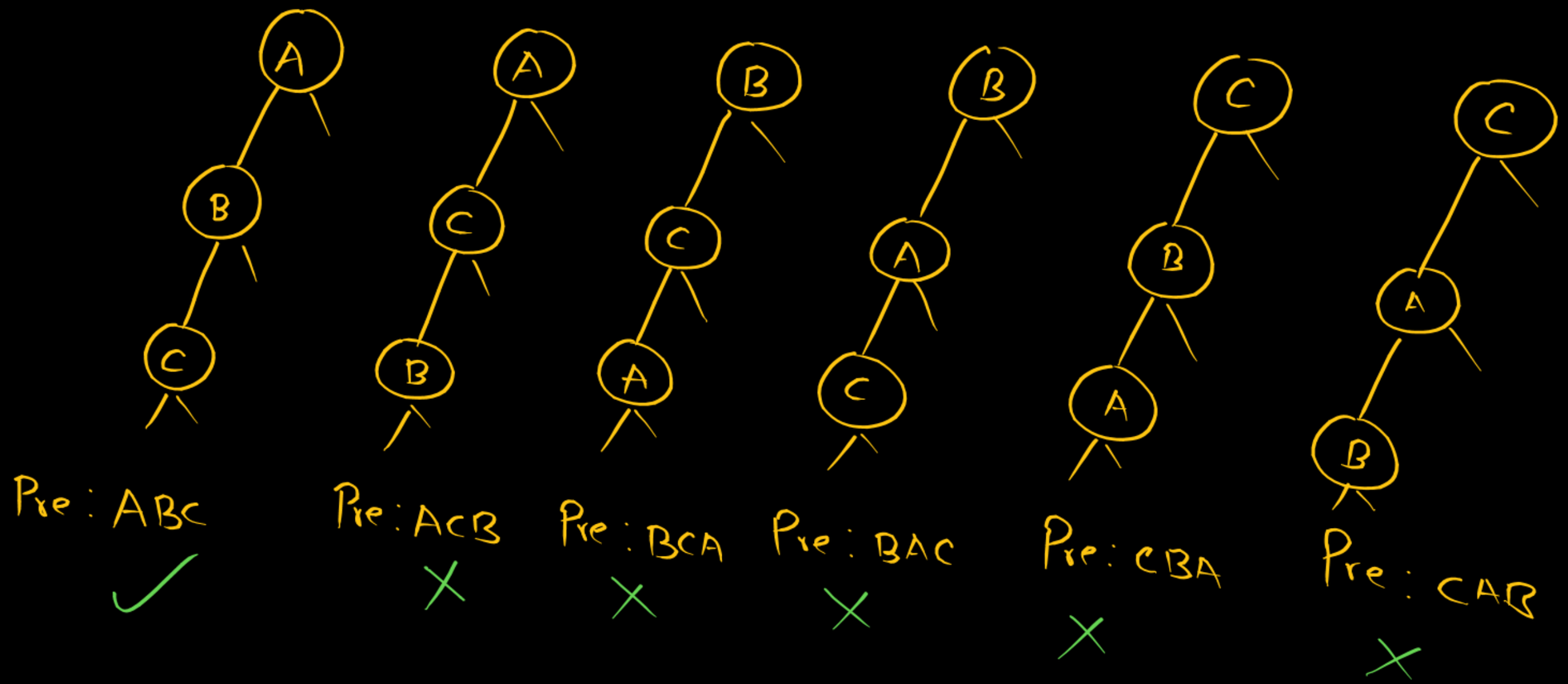
$= 30$

labelled binary trees with n distinct keys $= \frac{{}^{2n}C_n}{n+1} \times n!$

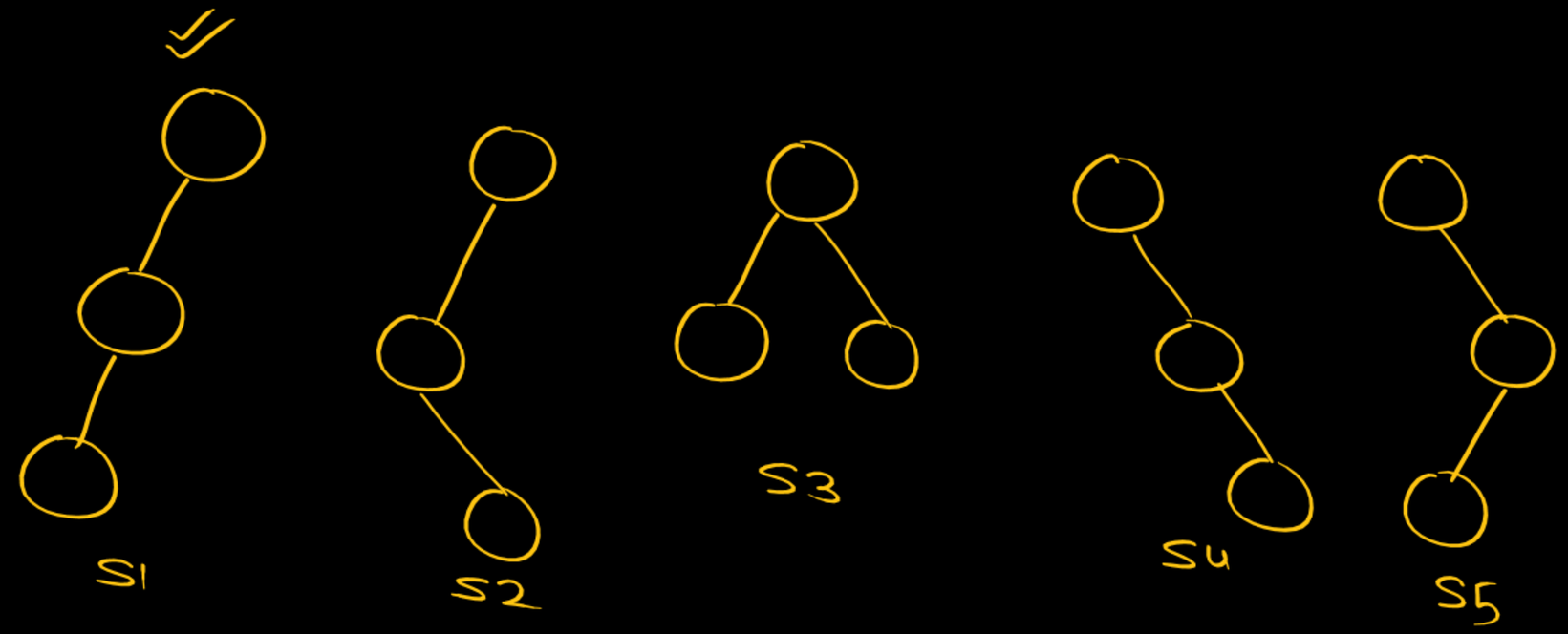
binary trees possible with preorder ABC.



For every structure 1 such binary tree is possible



binary trees possible with preorder ABC.

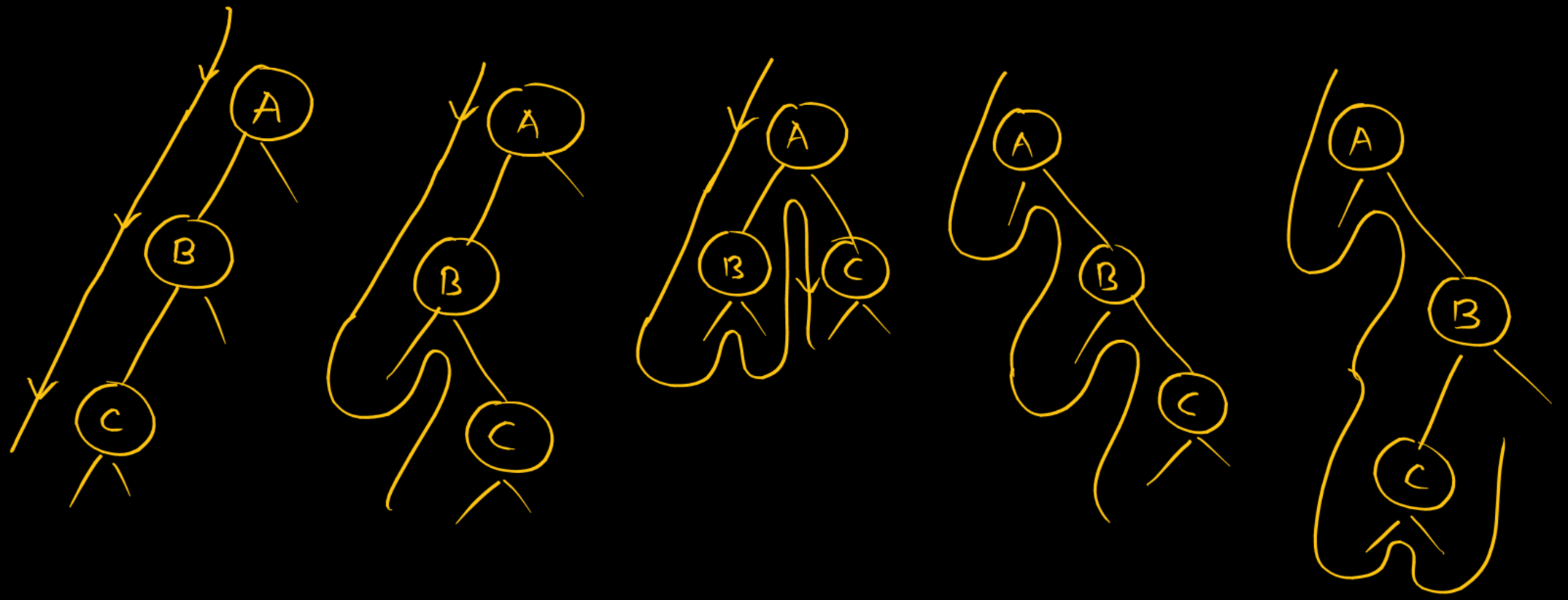


⇒ # binary trees with preorder as ABC = # binary tree structure with 3 nodes

binary trees with a given preorder of n length $= \frac{{}^{2n}C_n}{n+1}$

Q # binary trees with preorder DABCE $= \frac{{}^{10}C_5}{6}$

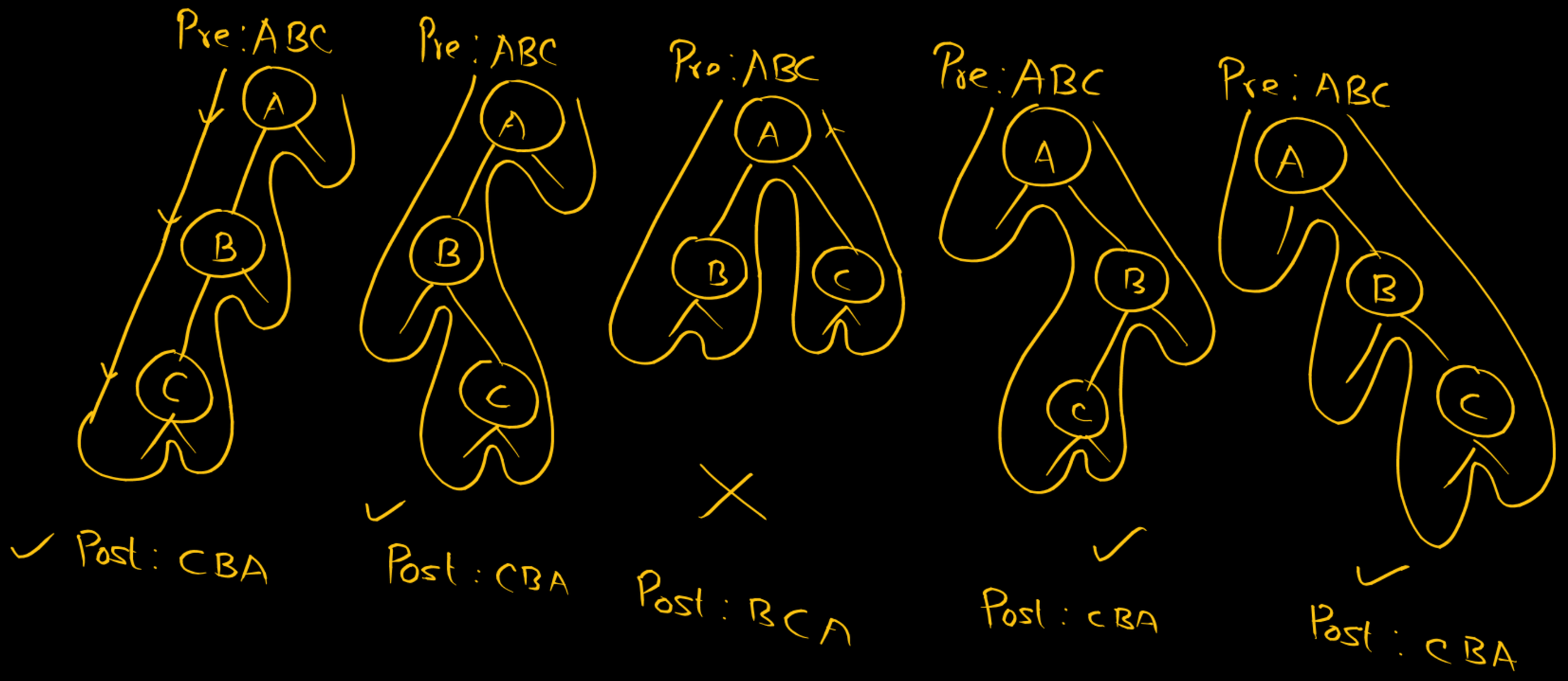
Pre : ABC →



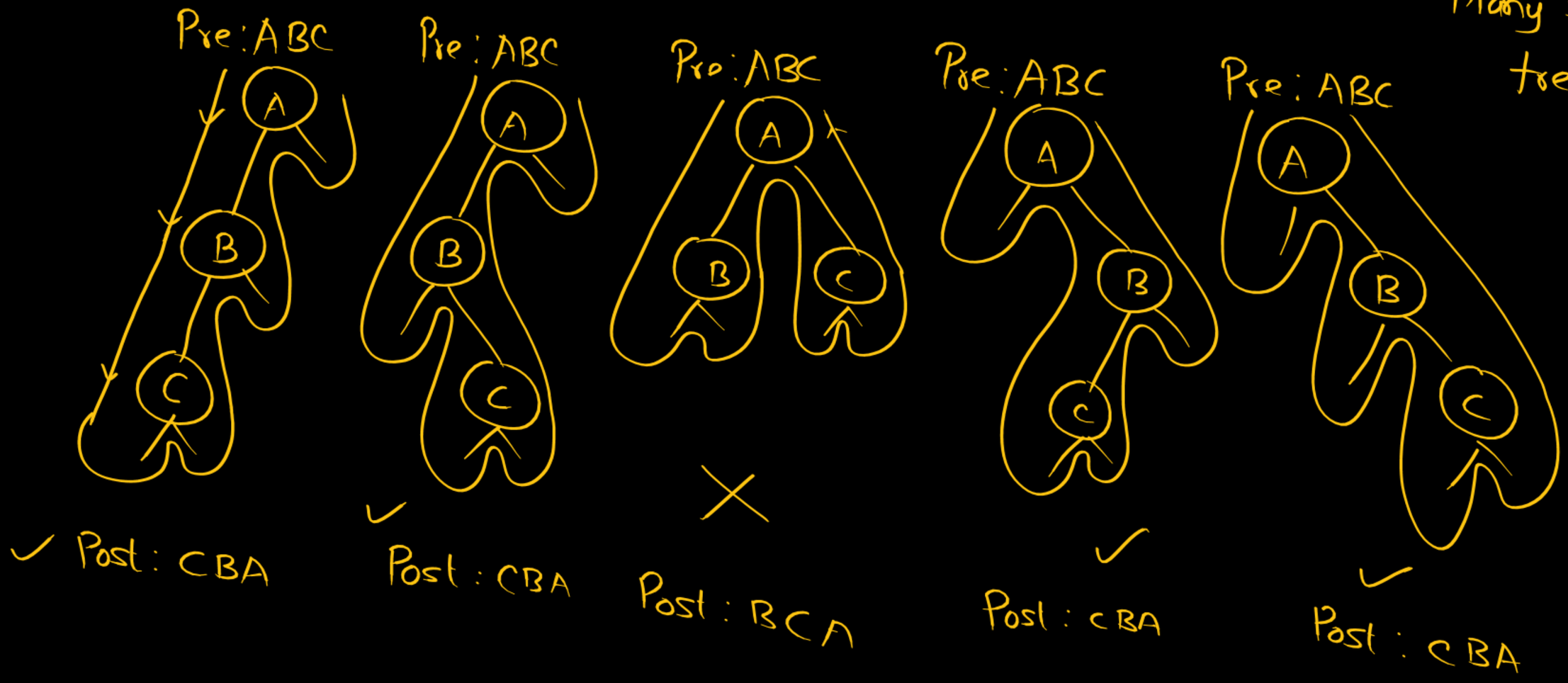
Given postorder (length n), no. of binary tree = $\frac{2nC_n}{n+1}$

With any 1 given traversal (Pre/In/Post), the no. of binary tree possible = $\frac{2nC_n}{n+1}$

binary trees possible with preorder ABC & postorder CBA



binary trees possible with preorder ABC & postorder CBA



⇒ Many such trees

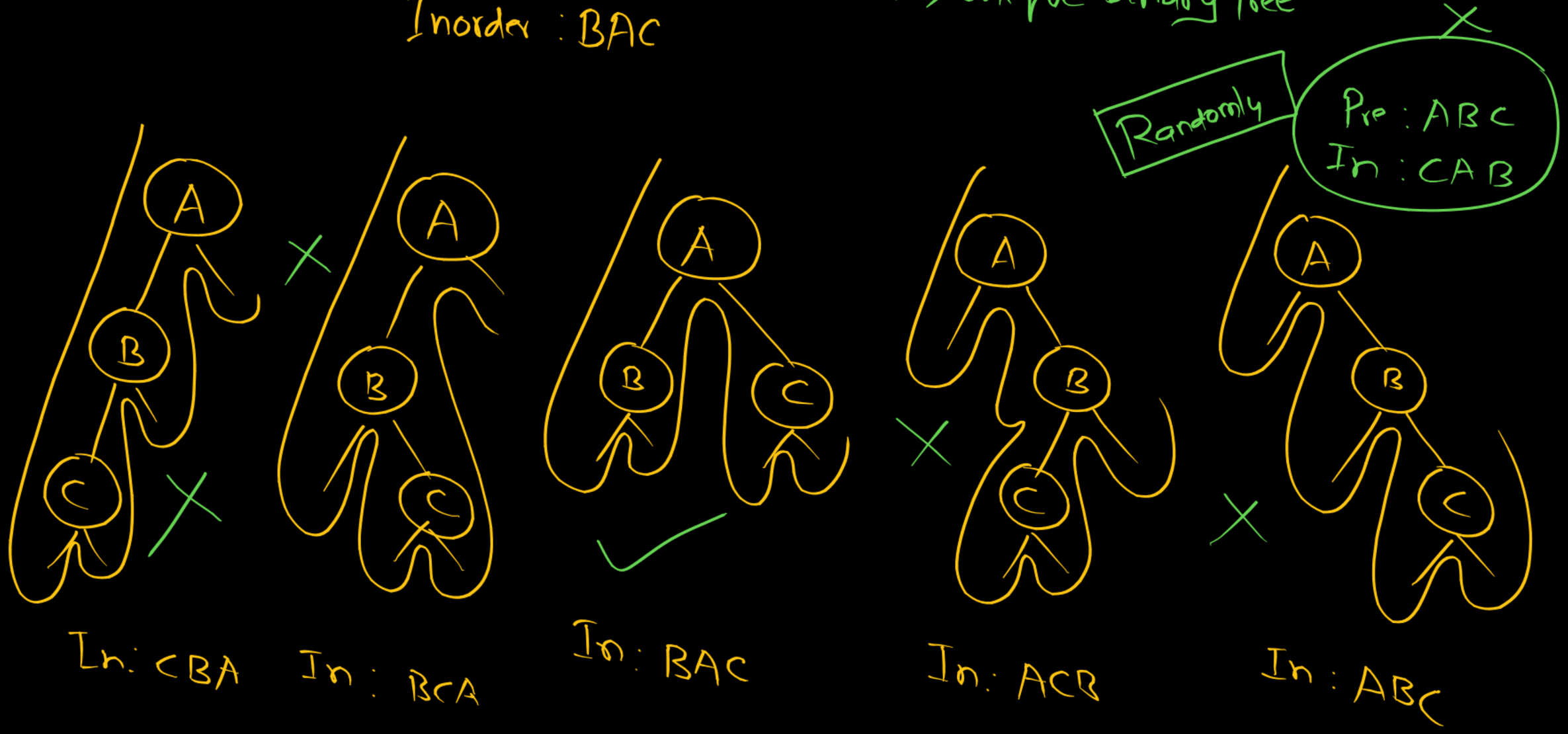
Pre : ABC
Post : BAC
→ No binary tree

Q Binary tree → Pre: / Post: } Given ⇒ Many trees

Q) Randomly Pre: / Post: } Given ⇒ Zero or more

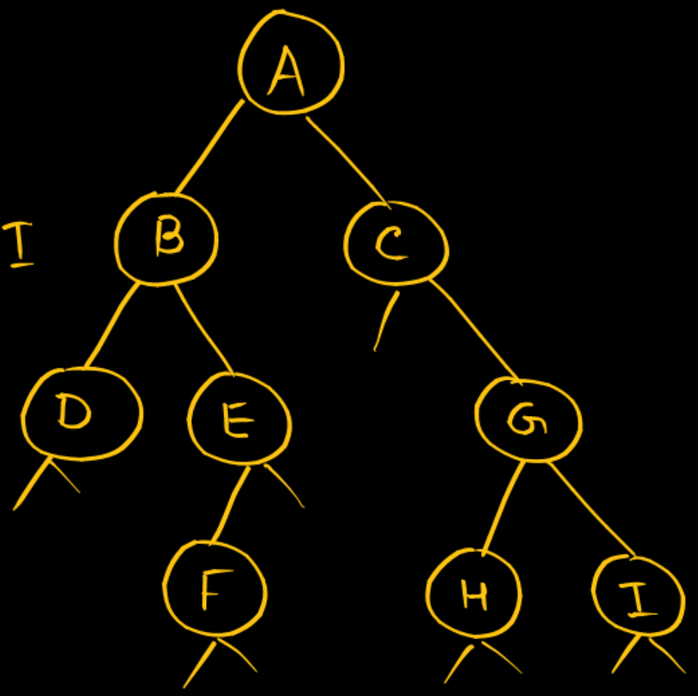
binary trees with Preorder : ABC &
Inorder : BAC

⇒ unique binary tree



Pre : A B D E F C G H I

In : D B F E A C H G I



Pre : A B D E F C G H I

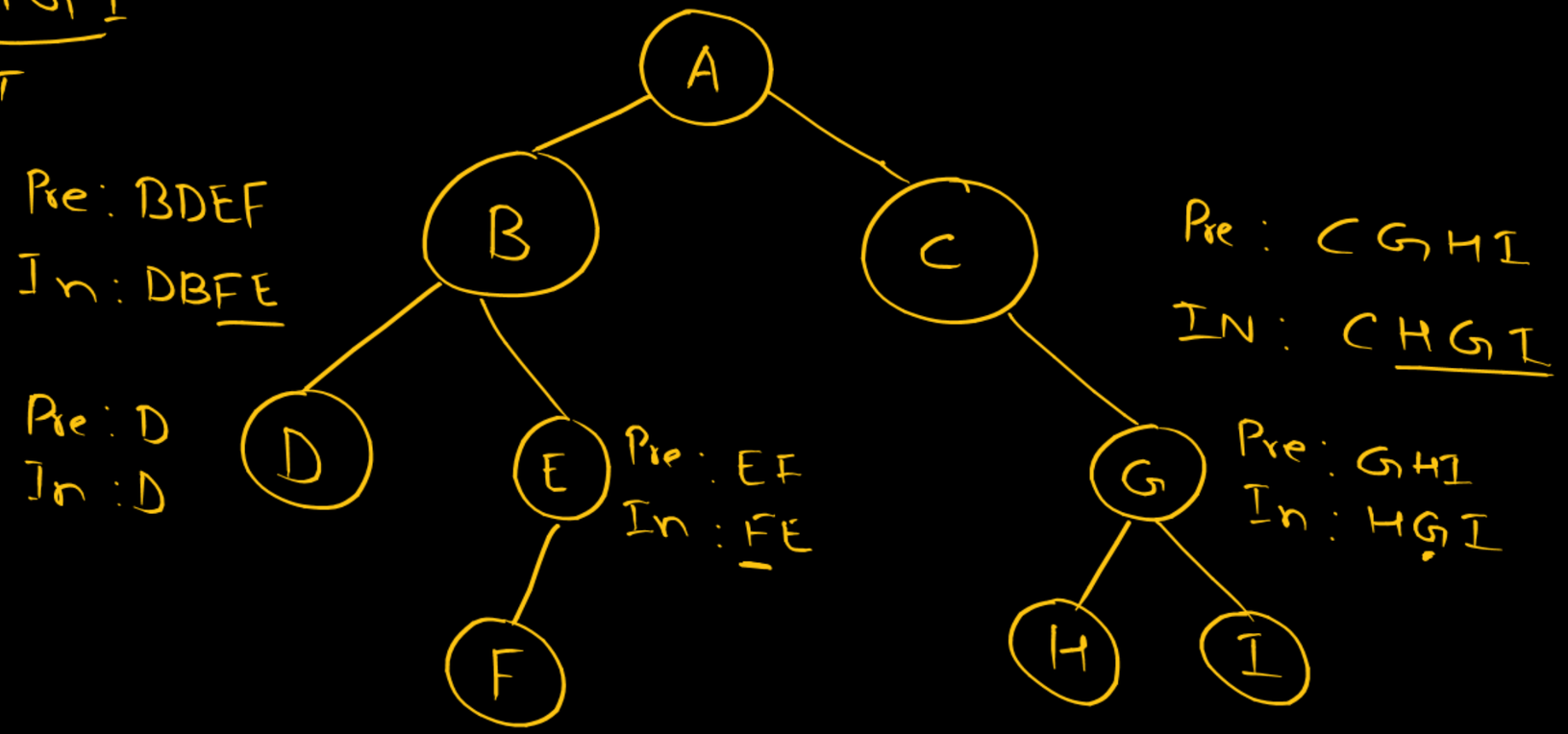
In : D B F E A C H G I

LT RT

Root, L_T, R_T

L_T, Root, R_T

Pankaj sir PW



Pre: BDEF

In: DBFE

Pre: D

In: D

Pre: EF

In: FE

Pre: CGHI

IN: CHGI

Pre: GHI

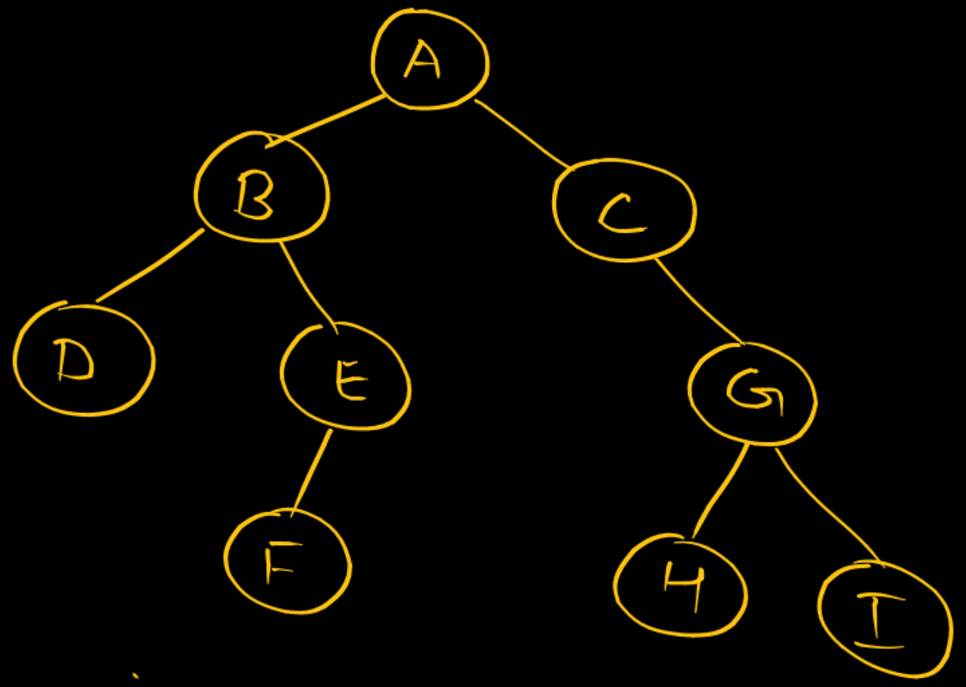
In: HGI

Pre : A B D E F C G H I

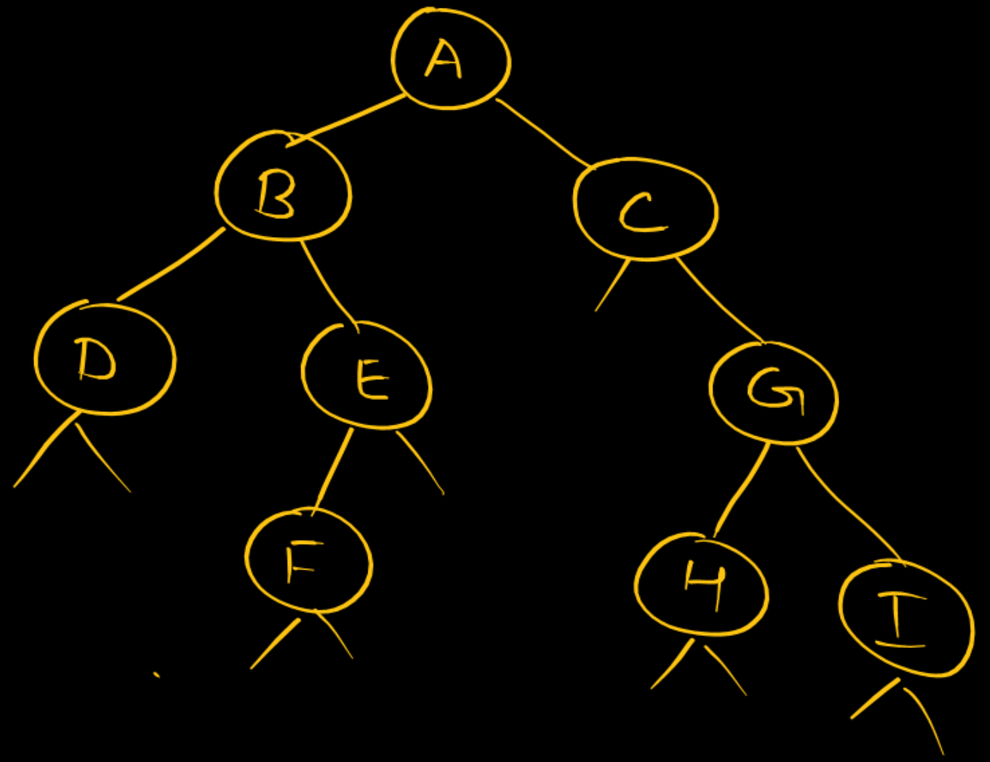
In : D B F E A C H G I

Short - trick

t.me/PWbankrajsirP



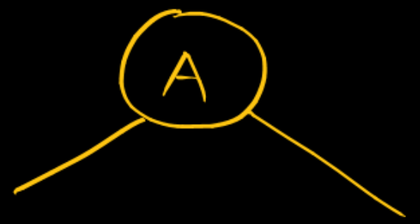
Post : DFEBHIGCA
In : DBFEACHGI



Post : DFEB HIGCA

In : DBFEA CHGI

LT RT



Post : DFEB

In : DBFE

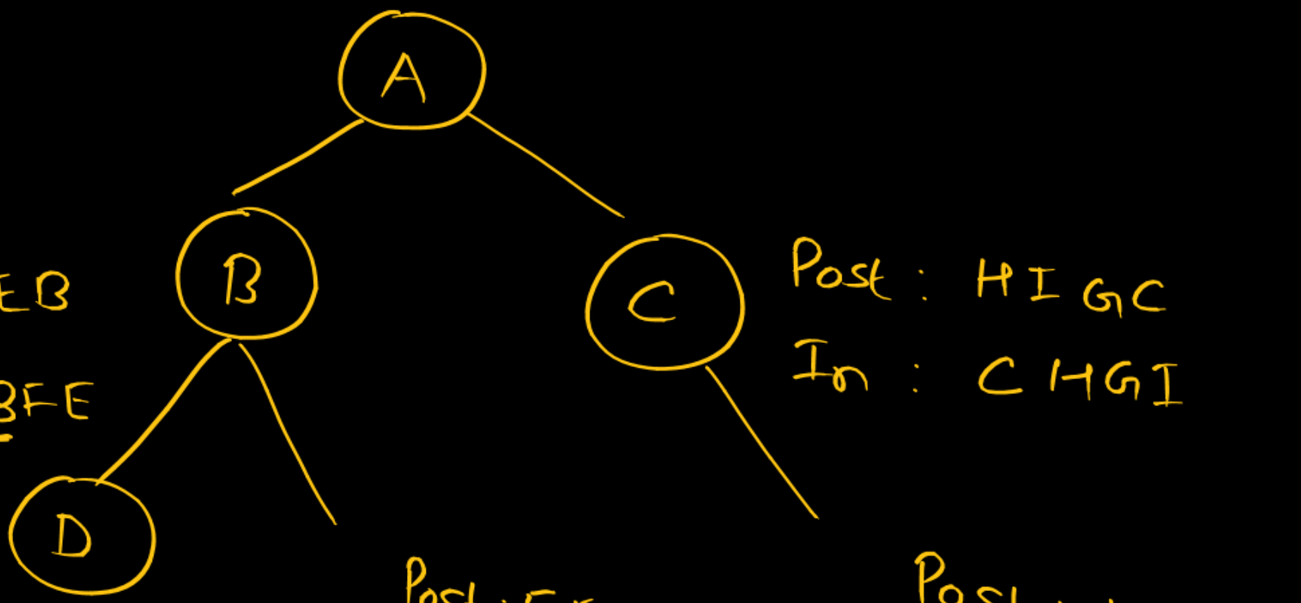
Post : HIGC

In : CHGI

Post : DFEB HIGCA

In : DBFEACHGI

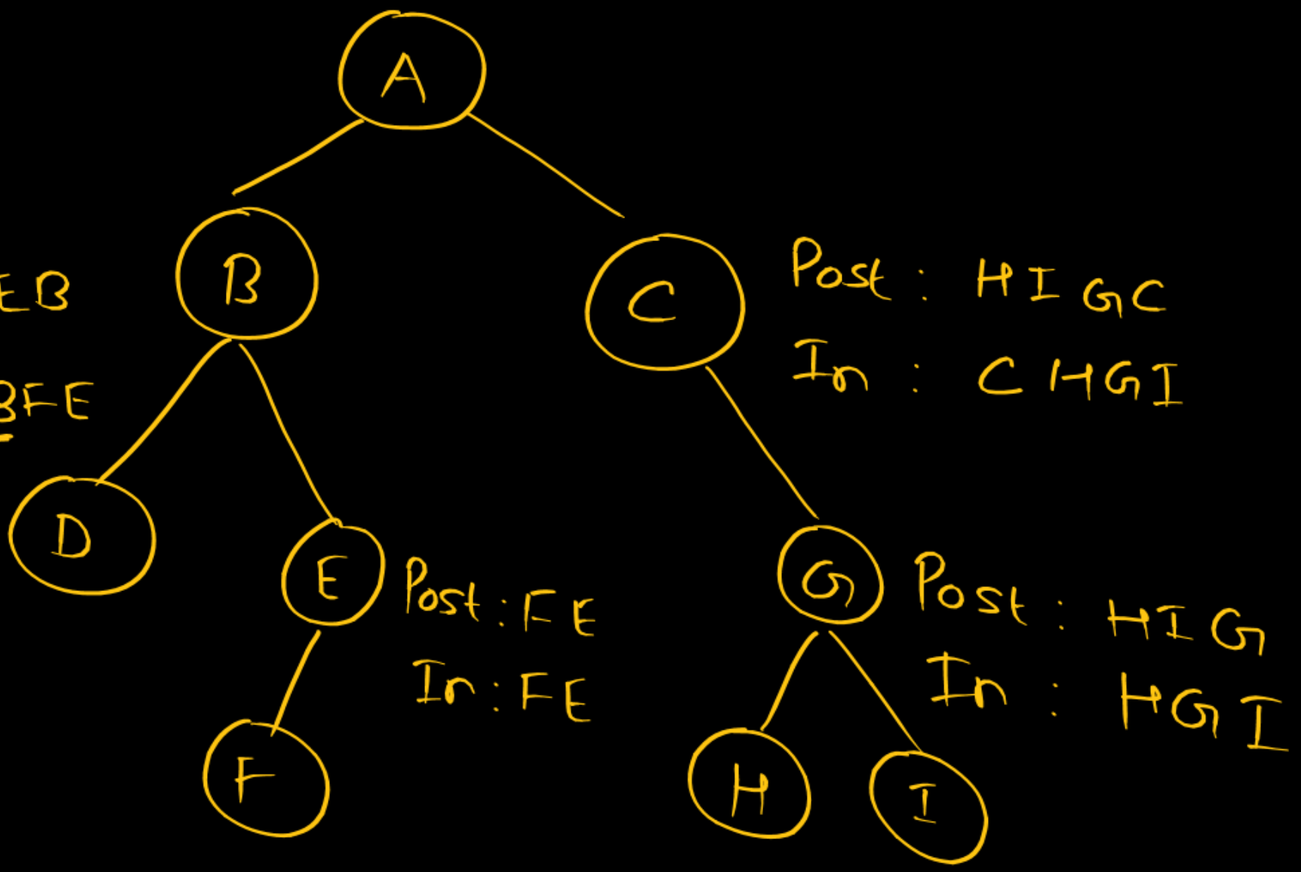
LT RT



Post : DFEB HIGCA

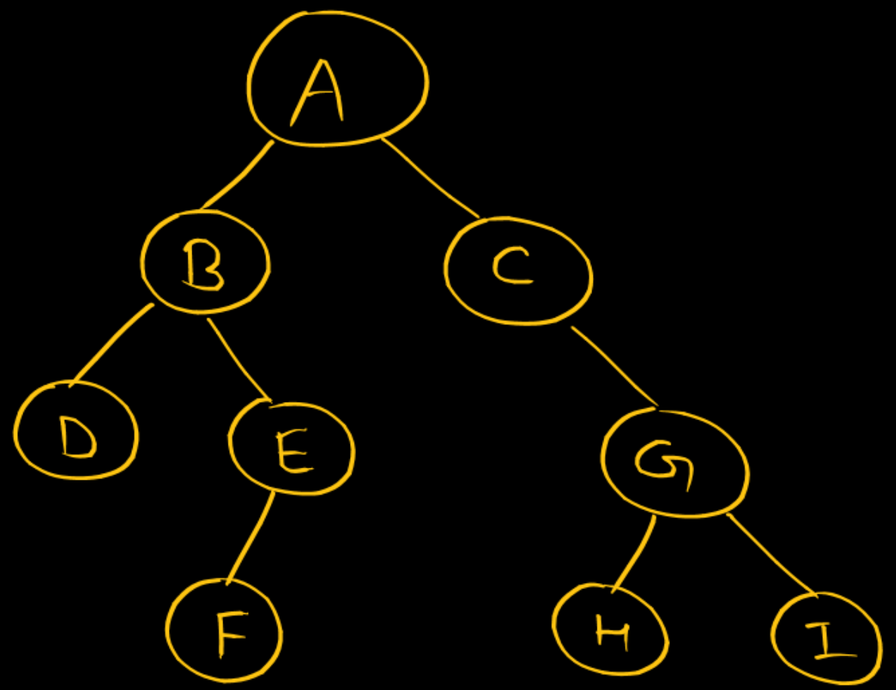
In : DBFEACHGI

LT RT



Post : D F E B H I G C A

In : D B F E A C H G I



binary Tree → 1 leaf node

↓

Every node has 0 or 1 child

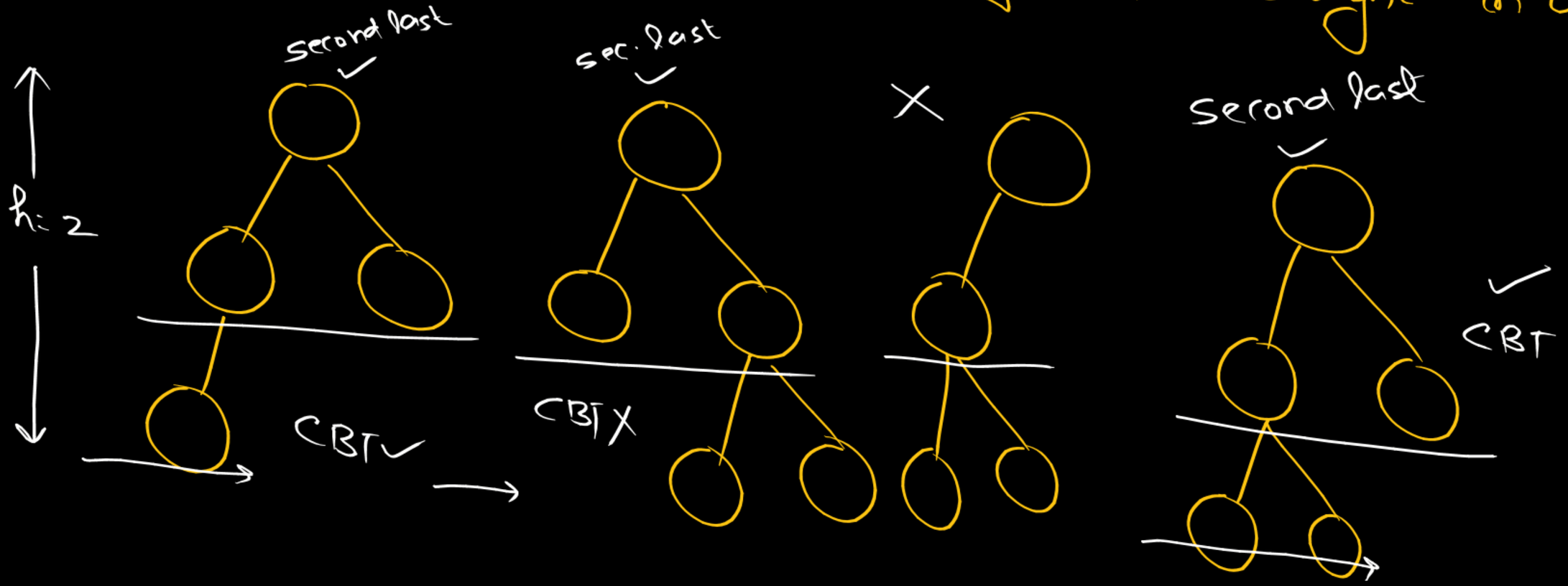
Q # binary tree with $h=3$ & 1 leaf node.

Q # binary tree with $h=7$ & 1 leaf node

H.W

Complete binary Tree

A CBT is a binary tree which is full upto second last level & nodes at last level are filled from left to right in order.

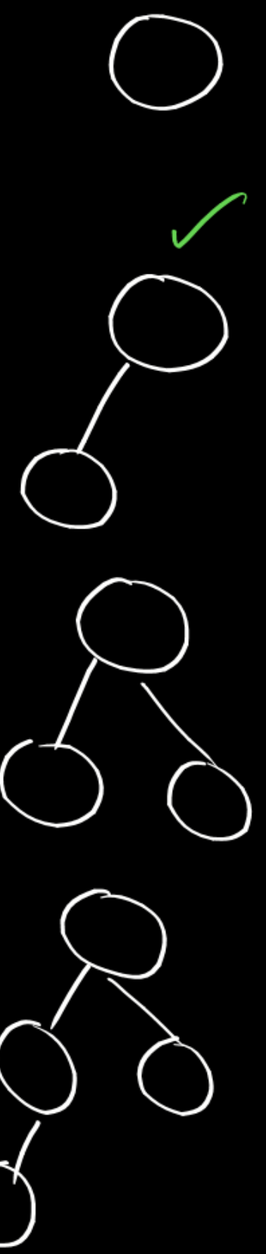


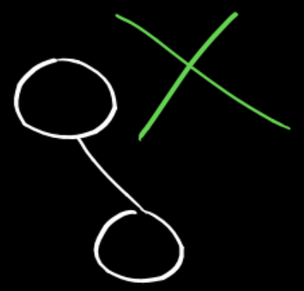
1) Structures of CBT with 1 node

2) Structures of CBT with 2 node

3) Structures of CBT with 3 nodes

4) Structure of CBT with 4 nodes





* The structure of a CBT with k nodes is always fix

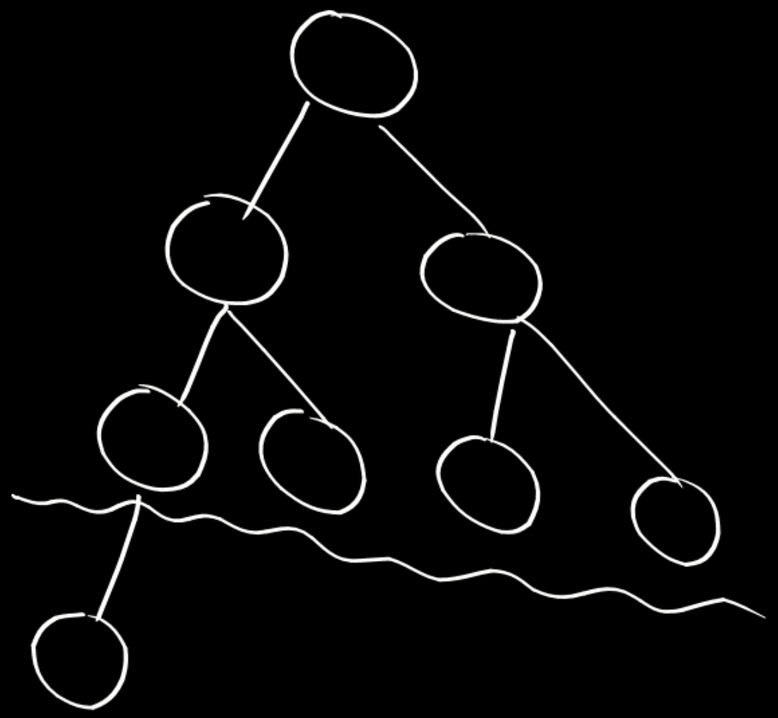
CBT

* Max. no. of nodes in a CBT of h height = $2^{h+1} - 1$

* Min no. of nodes in a CBT of h height.

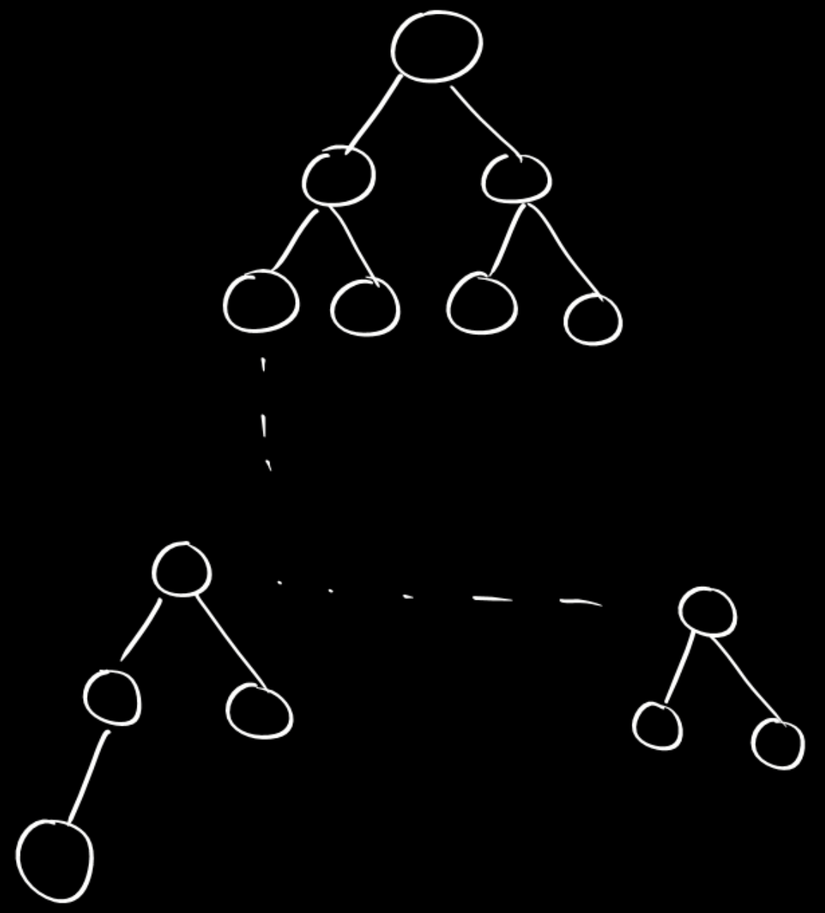
h = 3

Why 1 node?



level	
0	
1	
2	

$$N = (2^0 + 2^1 + 2^2) + 1$$
$$= 2^3 - 1 + 1$$
$$= 2^3$$



level	# nodes
0	2^0
1	2^1
2	2^2
⋮	
$h-2$	2^{h-2}
$h-1$	2^{h-1}
h	$\underline{1}$

$$n_{min} = \underbrace{\left(2^0 + 2^1 + 2^2 + \cdots + 2^{h-1}\right)}_{h \text{ term G.P}} + \underline{1} \longrightarrow \left(\frac{2^h - 1}{2 - 1}\right) + 1 = 2^h - 1 + 1 = 2^h$$

$$2^h \leq n \leq 2^{h+1} - 1$$

$x \leq 10$ && $x \geq 3$

$2^h \leq n$

$\log 2^h \leq \log n$

$h \log 2 \leq \log n$

$h \leq \frac{\log n}{\log 2}$

$h \leq \log_2 n$ — (1)

$n \leq 2^{h+1} - 1$

$(n+1) \leq 2^{h+1}$

$\log(n+1) \leq (h+1) \log 2$

$(h+1) \log 2 \geq \log(n+1)$

$(h+1) \geq \log_2(n+1)$

$h \geq \log_2(n+1) - 1$ — (2)

$\Rightarrow$

$3 \leq x \leq 10$

$\log_2(n+1) - 1 \leq h \leq \log_2 n$

THANK - YOU