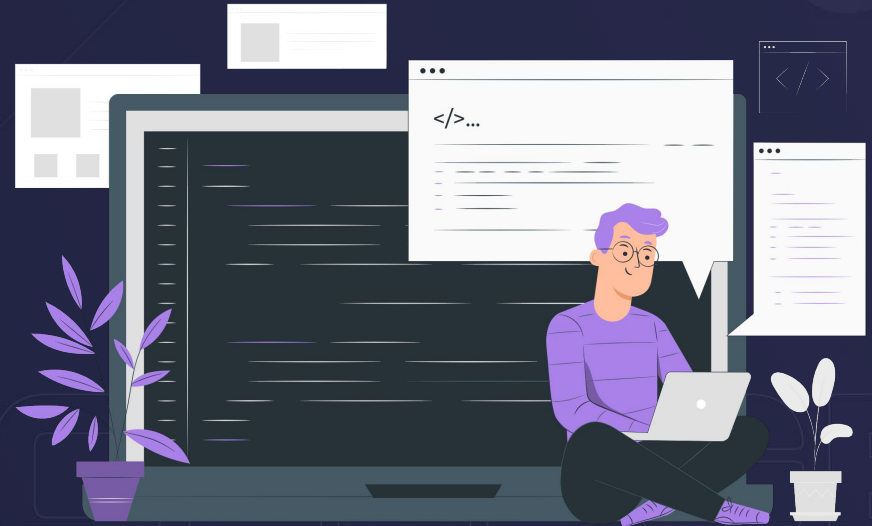


Lecture 55

Binary Search Trees



Recap

- Trees
- Interview Problems on Trees

COLLEGE
WALLAH

Today's Checklist

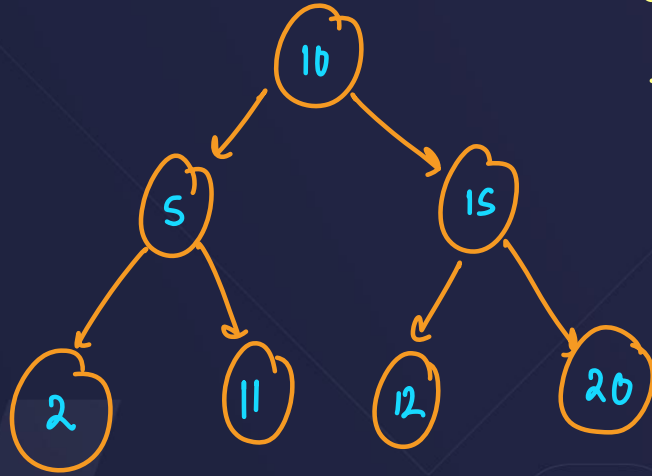
- Why Binary Search Tree?
- What is Binary Search Tree?
- Advantages
- Disadvantages
- Applications
- Insertion
- Traversal – Inorder, Preorder, Postorder
- Searching
- Practice problems on BST

COLLEGE
WALLAH

What and Why?

→ the entire LST
Every node to the left of a node should be smaller & Every node to the right should be greater.

→ applies for all nodes



↓
invalid BST

Ques : Can a BST contain duplicate elements ?

→ No, But

$LST < root \rightarrow val < RST$



$LST \leq root \rightarrow val < RST$



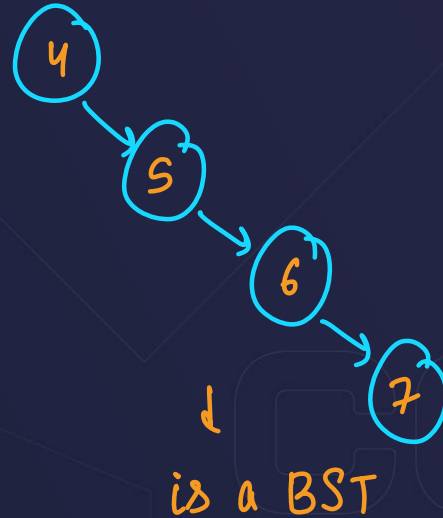
Advantages

- Efficient Searching
- Efficient insertion and deletion
- Usage in implementation of other data structures like **sets, maps, priority queues** etc.

COLLEGE
WALLAH

Disadvantages

- Lack of support for range queries
- Not that efficient in case of Unbalanced Trees



Applications

- Phonebook
- Dictionary
- Stock market analysis

COLLEGE
WALLAH

Concept Builder:

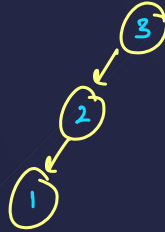
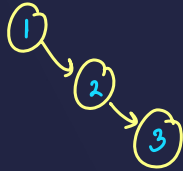
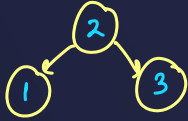
Q,, You have 3 nodes with diff. values, for ex $\rightarrow 1, 2, 3$.

How many unique Binary Trees can be formed?



Q// Same ques, but in case of BST what will happen?

1, 2, 3

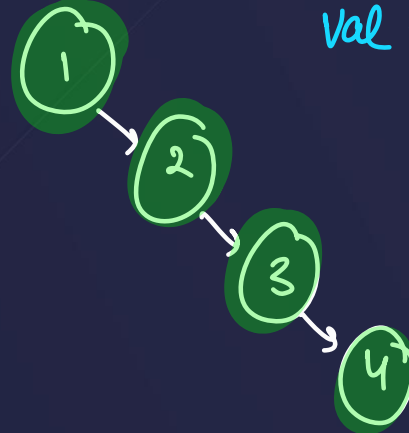


Search in a BST

[LeetCode 700]



val = 8



if (root->val == val) return root

if (root->val > val) go left

if (root->val < val) go right

Search in a BST

[LeetCode 700]

T.C.

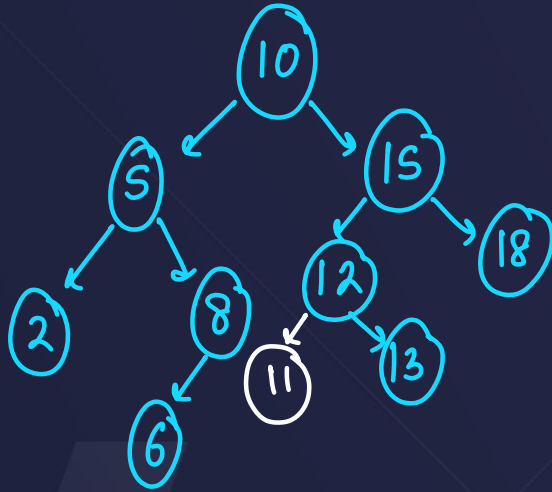
- Best Case = $O(\log n) / O(h)$
(balanced tree)
- Worst Case = $O(n) / O(n)$
- Avg. Case = $O(h)$

S.C.

Same as T.C.

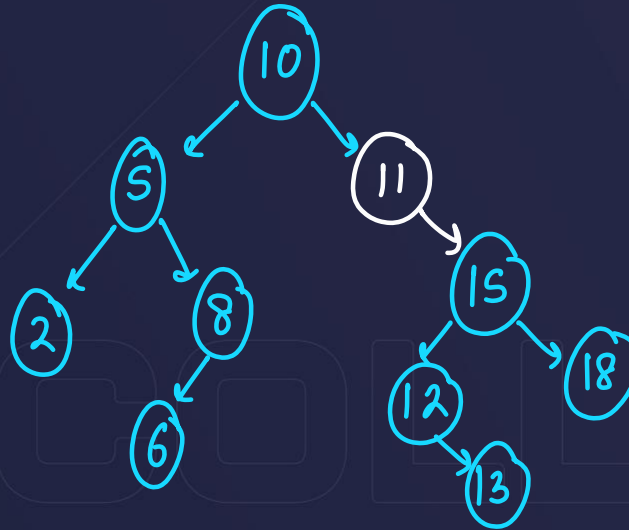
Insert into a BST

[LeetCode 701]



Valid

val = 11

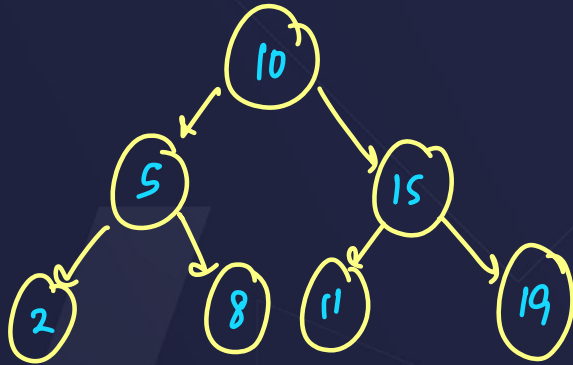


Valid

Traversal (Same as BT)

↓

Preorder, **Inorder**, Post Order



Pre : 10 5 2 8 15 11 19

Post : 2 8 5 11 19 15 10

inorder of BST is
SORTED

WVIMP
↑

** Inorder : (Left Root Right)

2 5 8 10 11 15 19

↓
sorted

COLLEGE
WALLAH

MCQ-1

Consider a binary search tree with n nodes.

What is the maximum possible height of the tree?

- A. ~~$O(n)$~~ n
- B. ~~$O(\log n)$~~ $\log n$
- C. ~~$O(n \log n)$~~ $n \log n$
- D. ~~$O(\sqrt{n})$~~ \sqrt{n}

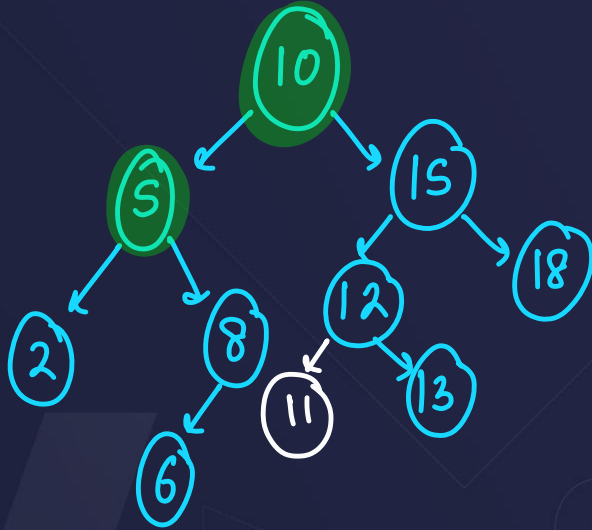
MCQ-2

Consider a binary search tree with n nodes. What is the minimum number of comparisons required to search for a value in the worst-case scenario?

- A. $O(1)$
- B. $O(\log n)$
- C. $O(n \log n)$
- ☒ D. $O(n)$

LCA of a BST

[LeetCode 235]



$$p = 8$$

$$q = 2$$

if (root->val > p->val && root->val < q->val) return root;

if (root->val > p->val && root->val > q->val

LCA of a BST

[LeetCode 235]



p, q
2, 4

root → val > max(LST) & & root → val < min(RST)

Validate BST

[LeetCode 98]

↓

You are given a binary tree (root).

You have to tell if the BT is a BST or not

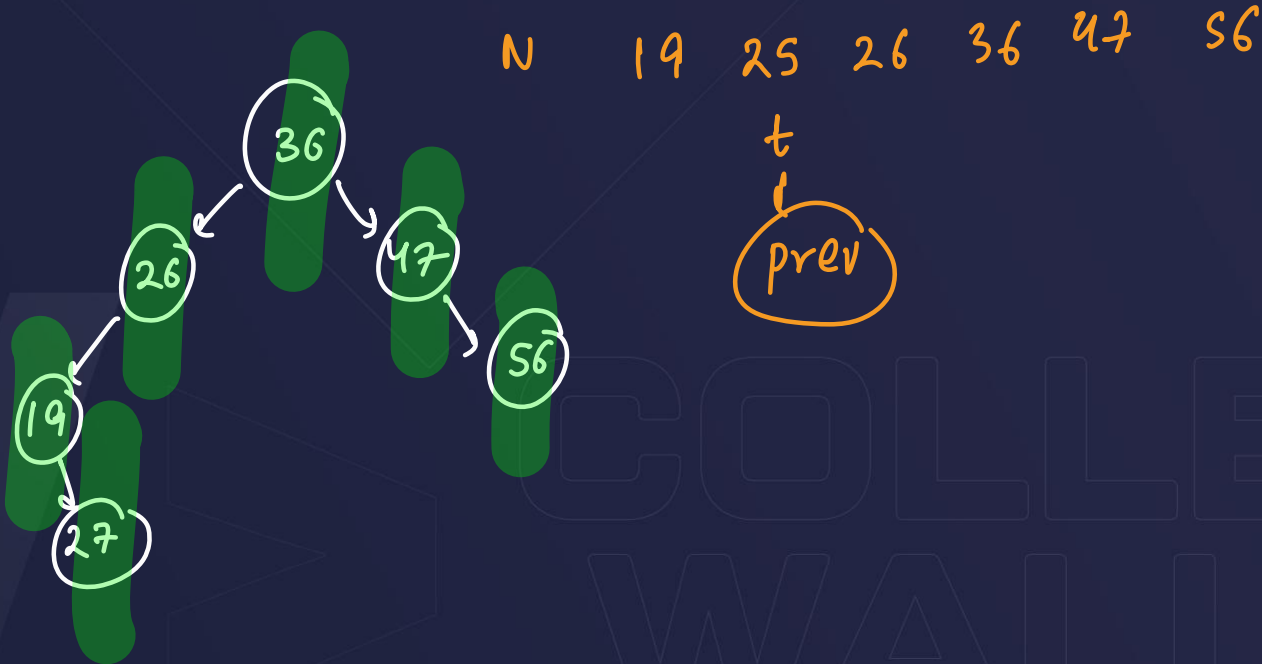
M-I: Use max of tree (Brute Force)

COLLEGE
WALLAH

Validate BST

[LeetCode 98]

Method-2 : Hint : Inorder Traversal



Ques : Given a BST, transform it into a **greater sum tree** where each node contains the sum of all nodes greater than that node. *+ original* **[Leetcode 1038]**



transform
→

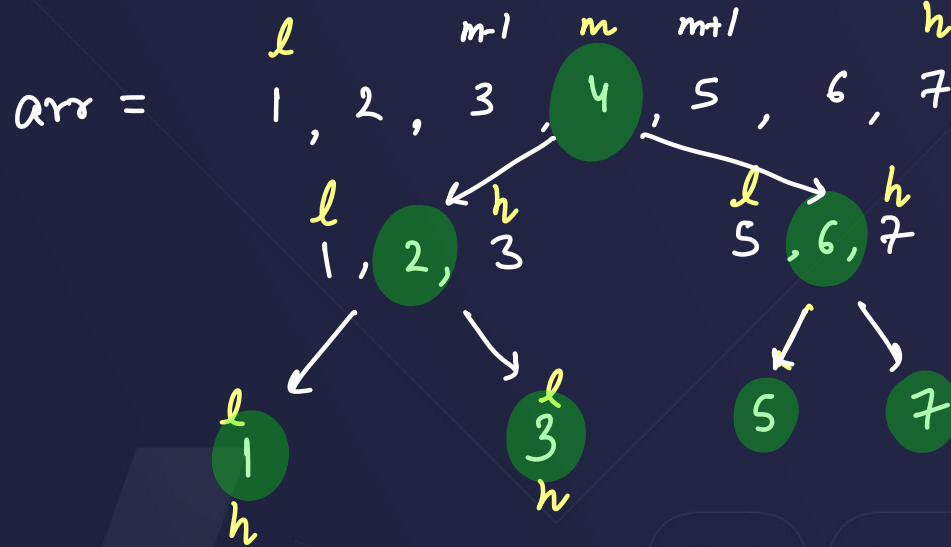


Sum = ∅ 1 11 15 18 20 21

1 2 3 4 5 6

Converted Sorted Array to Balanced BST

[LeetCode 108]



Ques : Given the preorder traversal of a binary search tree, construct the BST. [Leetcode 1008]

8 , 5 , 1 , 7 , 10 , 12



insert in BST → copy paste
↓
loop

Next Lecture

- Deletion of Nodes, more questions on BST

COLLEGE
WALLAH



▶ **THANK YOU** ◀

COLLEGE
WALLAH