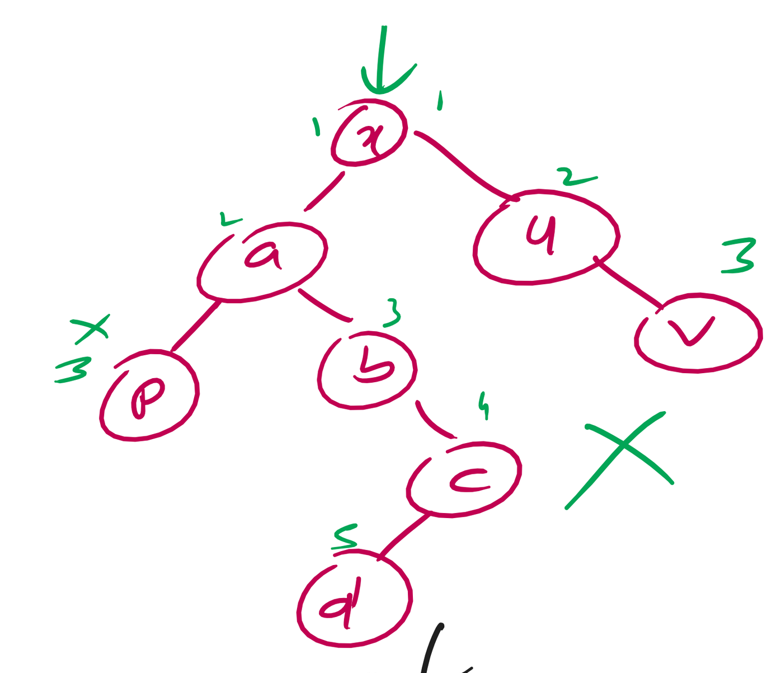
BST interview question :→

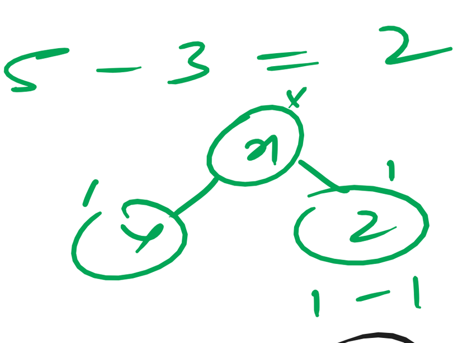
* Given a sorted array, convert the given array into a Balanced Binary Search Tree.

i/p : arr = {1, 2, 3, 4, 5, 6, 7, 8, 9}

Note : Balanced BST mean's left height − right height <= 1 (−1, 0, +1)

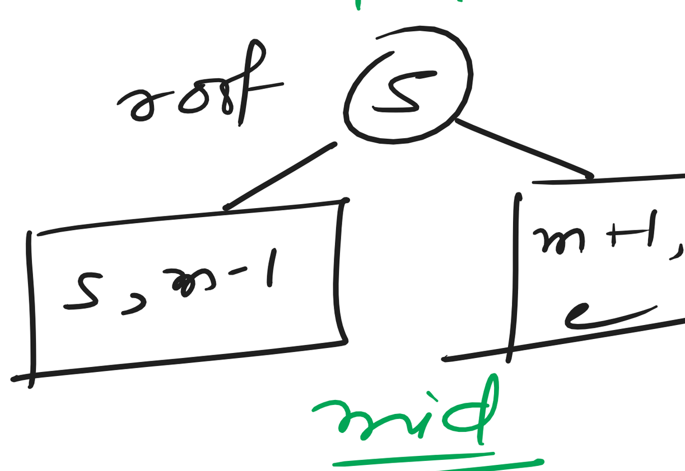


lh = 5
rh = 3
5 − 3 = 2 ✗

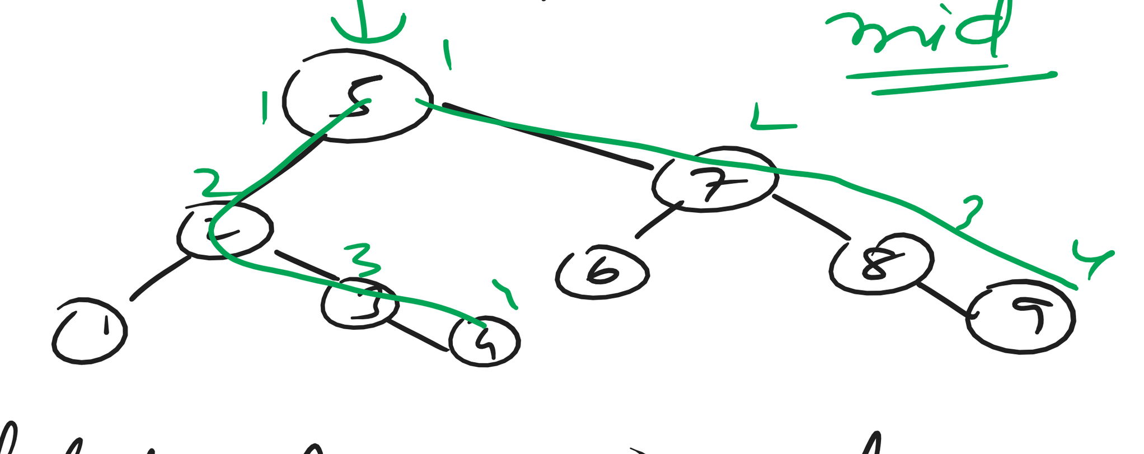


[1, 2, 3, 4, 5, 6, 7, 8, 9]
0 1 2 3 4 5 6 7 8
s m e

m = 8/2 = 4



4 − 4 = 0



* Important coding round questions on Collections Framework where very less time is given :→

** Competitive Coding :→

① Given an array of Strings, use a collection to remove duplicates. 2 − 3 mins max

TreeSet
Linked Hashset ✓ (Set) → Interface

*** TCS / Infosys / IBM / Accenture

Given an ArrayList of integers, use the Collection Iterator to traverse the List & remove data based on any given condition. 2 − 3 mins

Note : Iterator → It is an interface in Java used to iterate over collections. (auto in C++)

* What is Hashing, Hash Collision in an HashTable in Java?

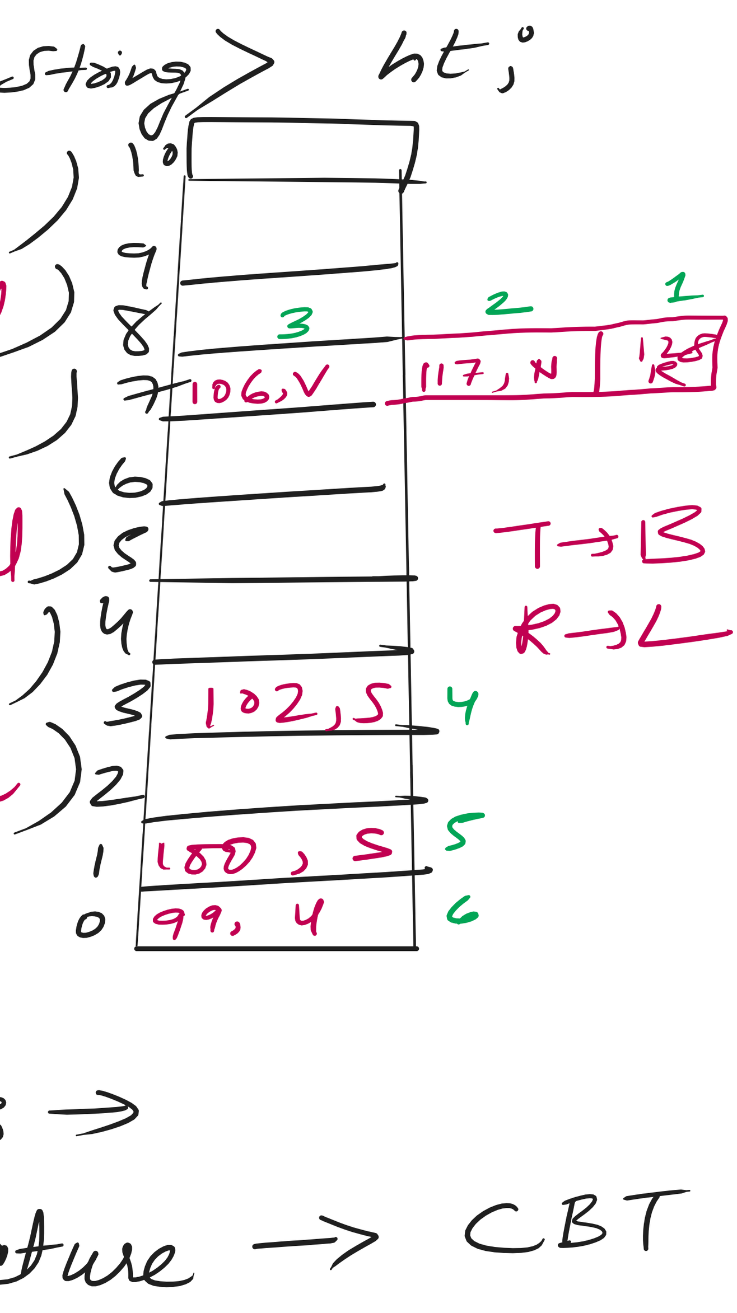
[C++] unordered_map unordered_set

HashTable <K, V>
↳ Array of Lists
* Initial size of HT is 11
* Load Factor is 0.75 or 75%

Hashing → Providing a unique code (Hashcode) to each element by the JVM.

When there is clash of hashcode it is called Hash Collision.

HashTable<Integer, String> ht;
ht.put(106, Vikas)
ht.put(117, Nikhil)
ht.put(128, Rajat)
ht.put(99, Ujjwal)
ht.put(100, Suraj)
ht.put(102, Sajin)
(11) ✓



Priority Queues :→

Heap Data Structure → CBT

Max Heap — Root
Min Heap — Root

TC O(1)

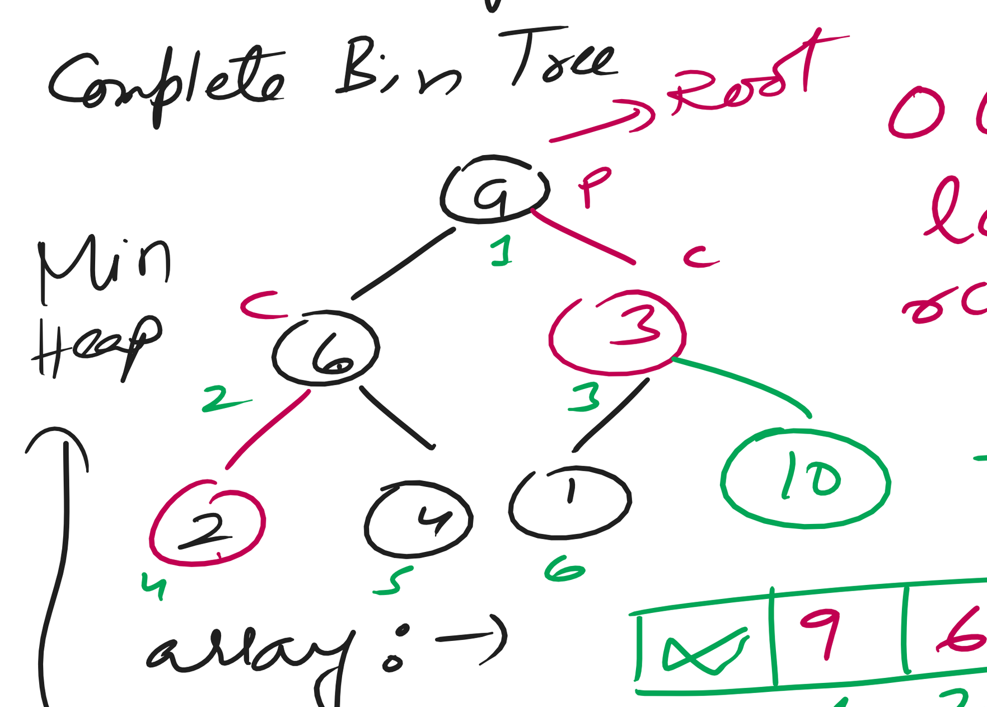
2	8	6	1

O(1)
max element O(n)

Max heap → 2, 6, 3, 4, 9, 1

Complete Bin Tree → Root O(1)

$lci = 2i$
$rci = 2i + 1$



heapify
heapSort

array :→

✗	9	6	3	2	4	1
0	1	2	3	4	5	6

Tips for Coding Rounds

(I) Accessing elements faster : O(1)
Index ← ArrayList, Priority Queue

(II) Inserting Elements faster : Linked Lists

(III) Removing duplicates :→ Set, HashSet, Linked HashSet

(IV) Two way operations → Deque

(V) Key-value pair → Hash Map ↓ Also Ascending order

(VI) Hashing <K, V> → HashTable

(VII) Traversal → Iterator