unacademy

Trees - Part VI

Course on Data Structure

Pankaj Sharma • Lesson 26 • Aug 6, 2024

unacademy

CS & IT Engineering

Data Structure

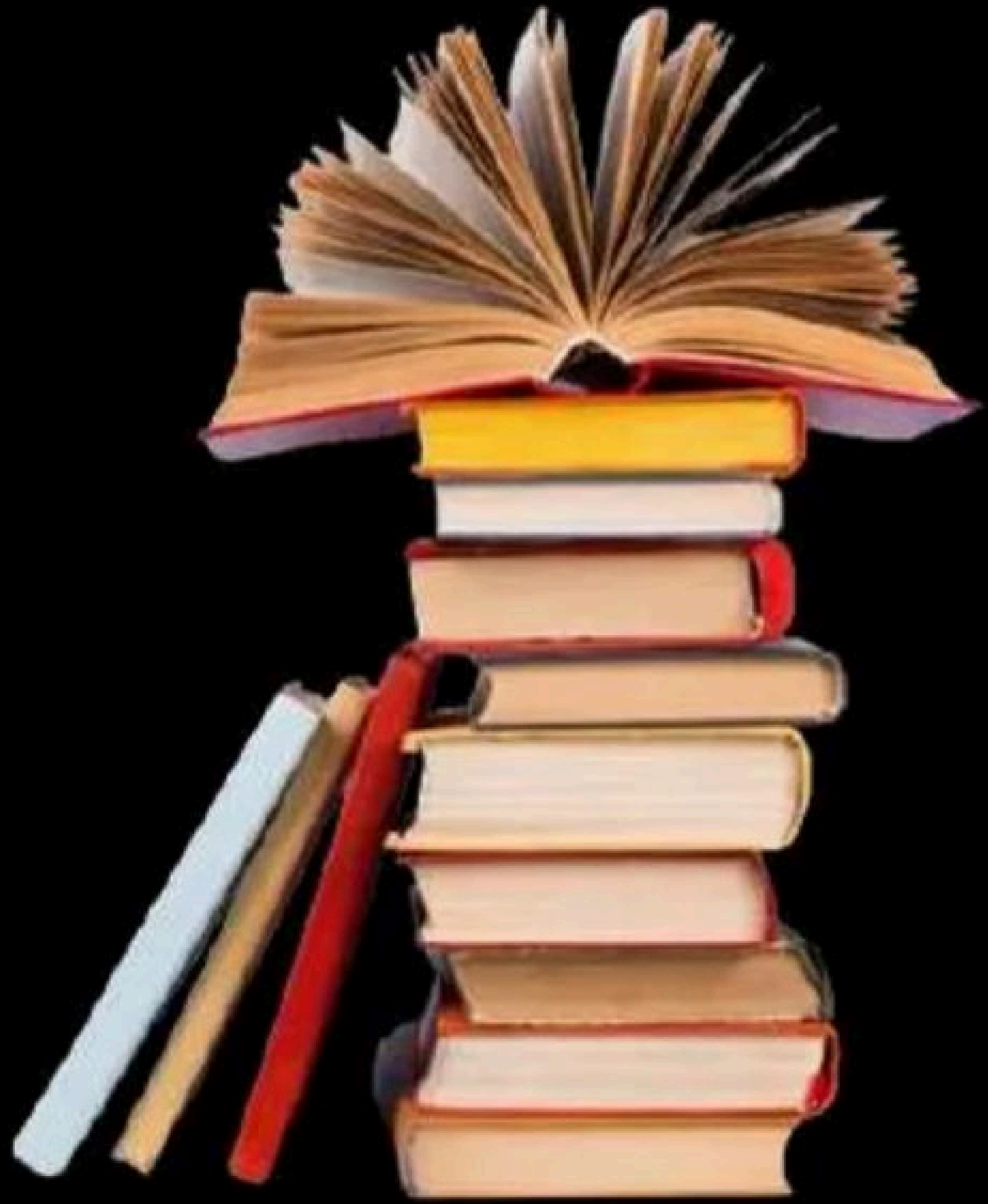
Tree

Lecture Number- 26

By- Pankaj Sir

Topics

to be covered



1. Tree-V

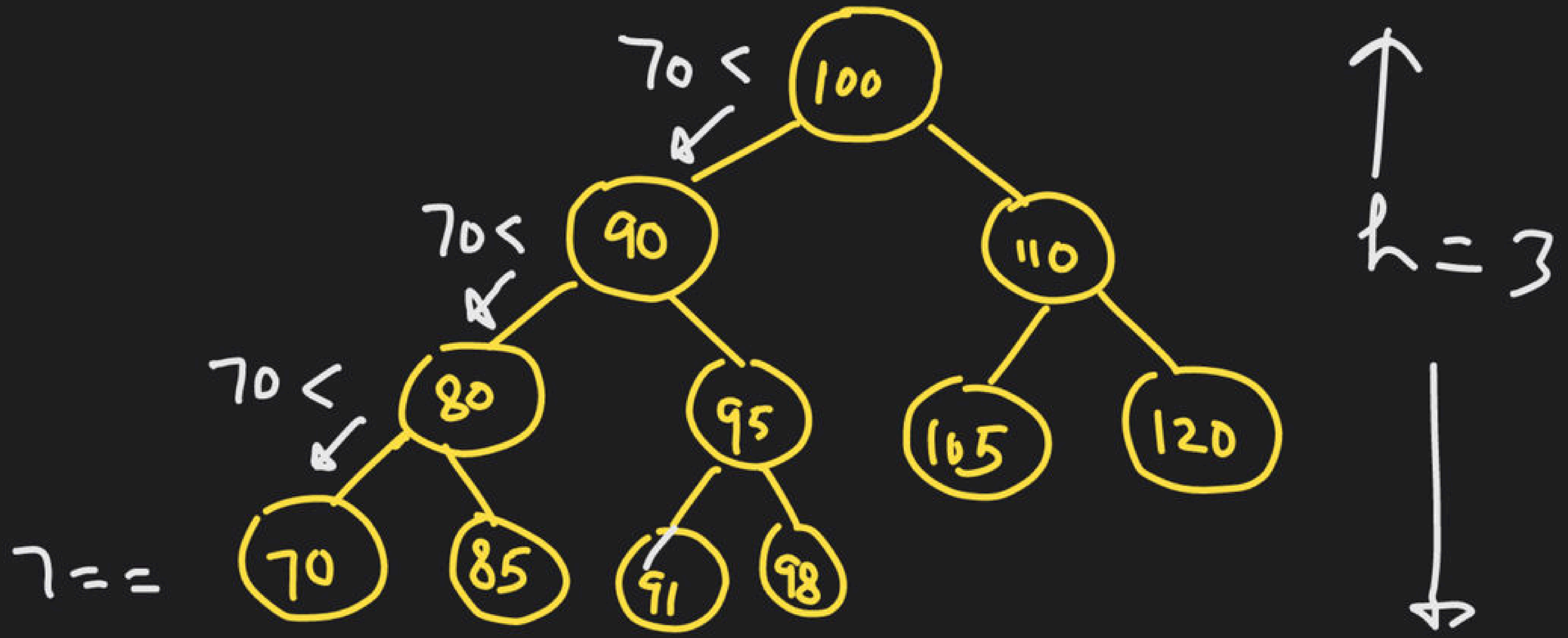
Search in a BST

Key = 70

#comp = 4

= h+1

= O(h)

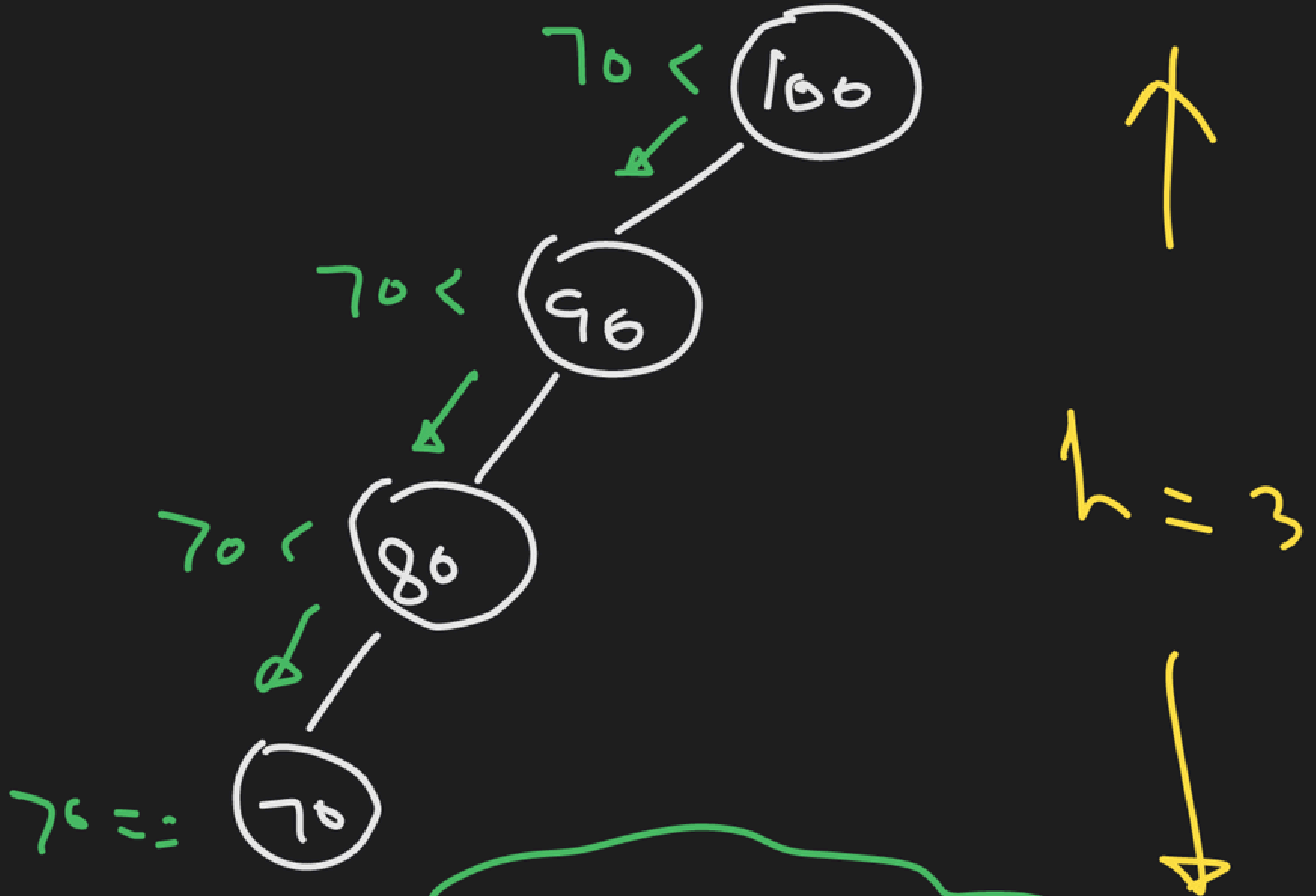


$\# \text{comp} = 4$

$= (h+1)$

$= O(h)$

$$\boxed{h \cong n}$$



Skewed tree

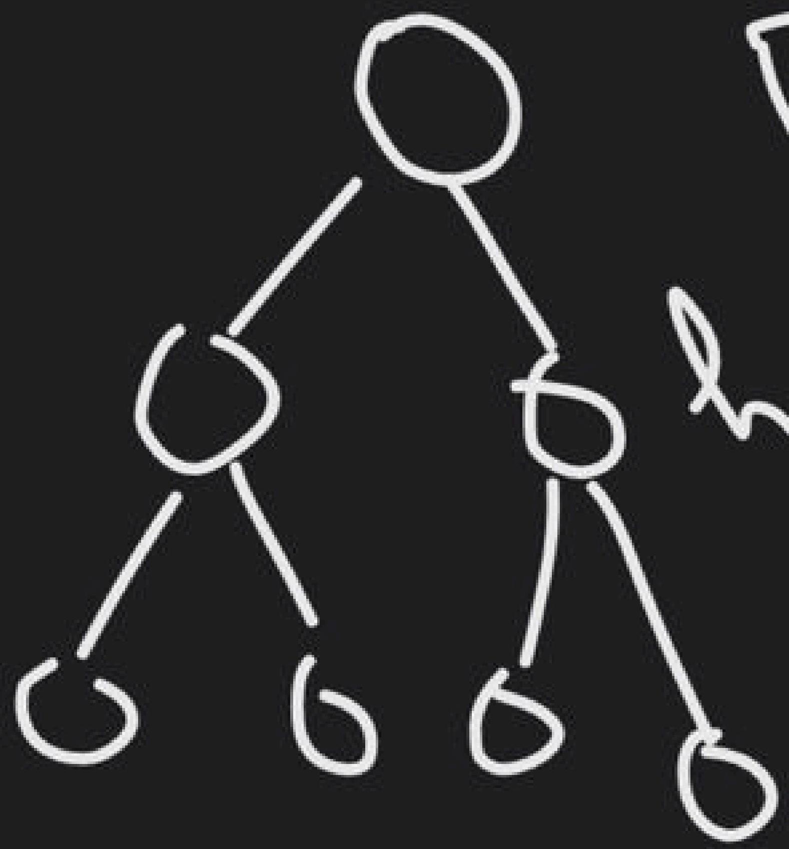
Search

↓

$O(h)$

FBT/CBT → $O(\log_2 n)$

Skewed → $O(n)$



FBT/CBT

$h \Rightarrow O(\log_2 n)$

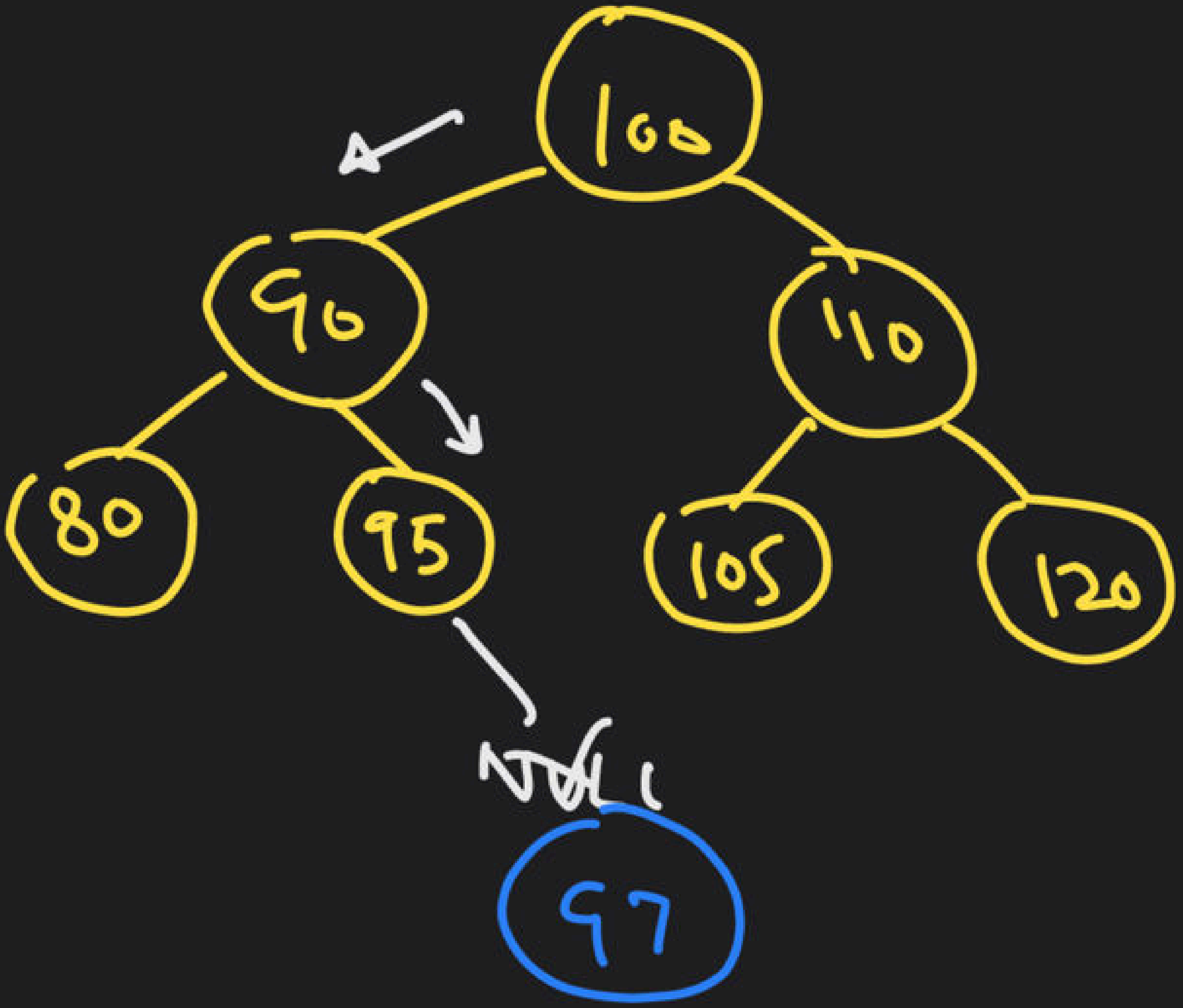
$\boxed{\text{Search} \Rightarrow O(n)}$

Insertion In a BST

insert : 97 → a) Search

Right of 95 Child

Complexity : O(n)



Deletion from a BST

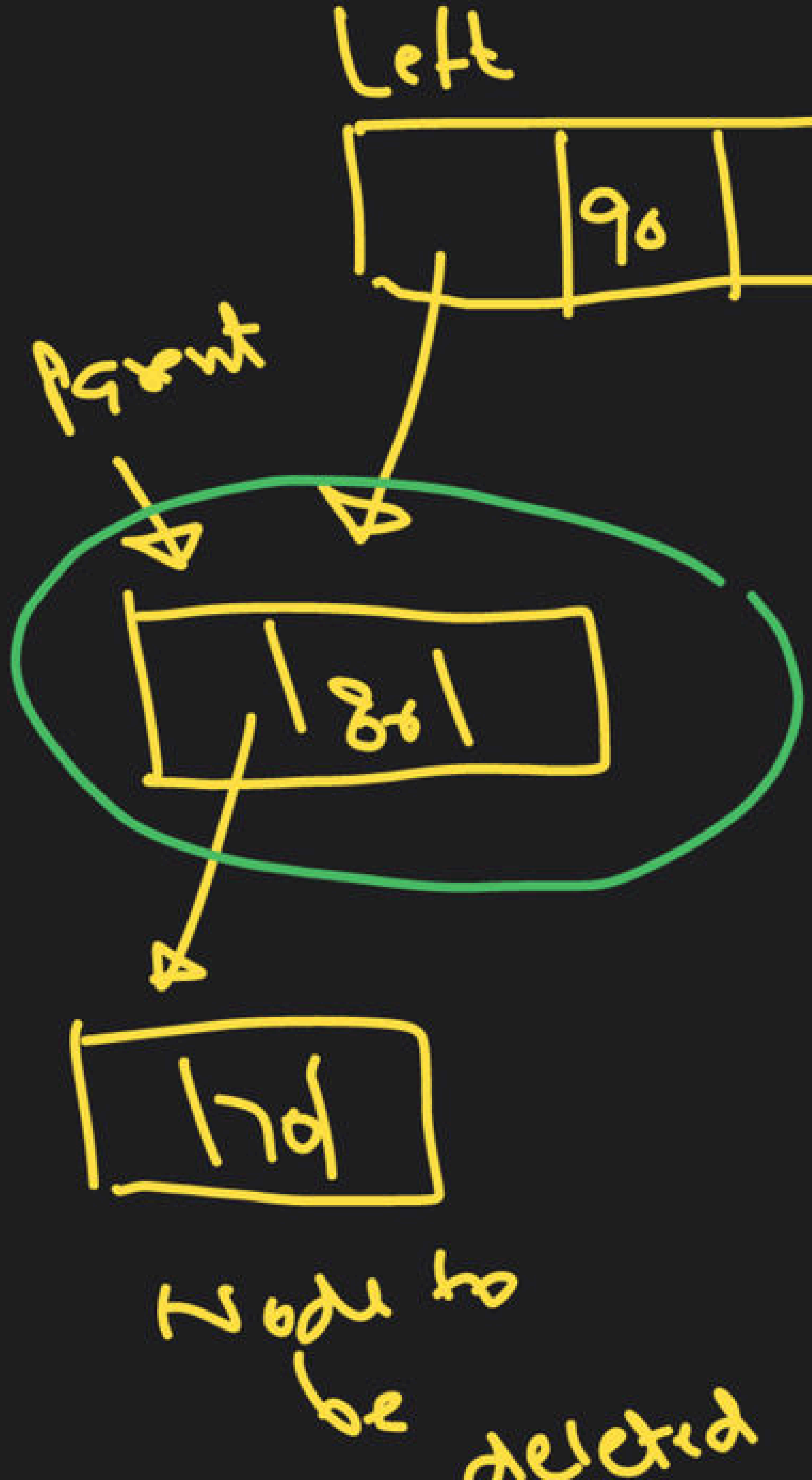
a) Delete a node with 0- child (leaf node)

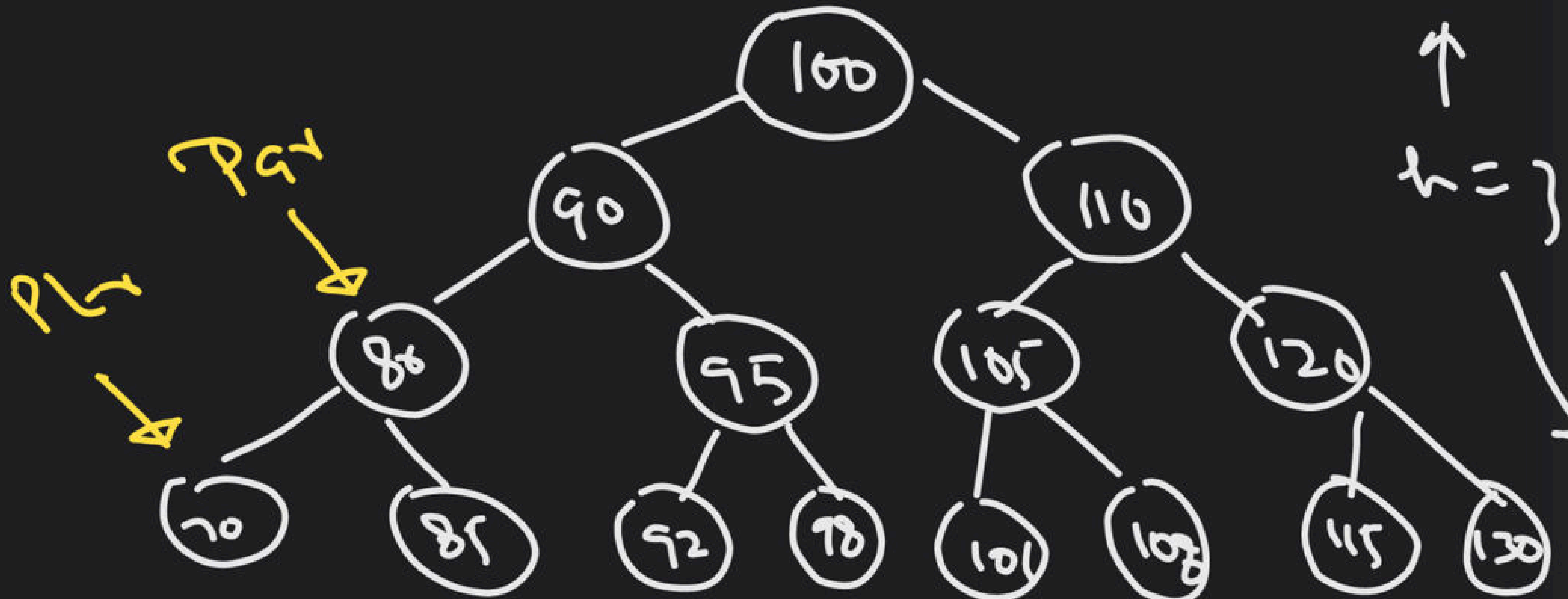
b) Delete a node with 1-child

c) Delete a node with 2-child

Deletion of leaf node

delete 70

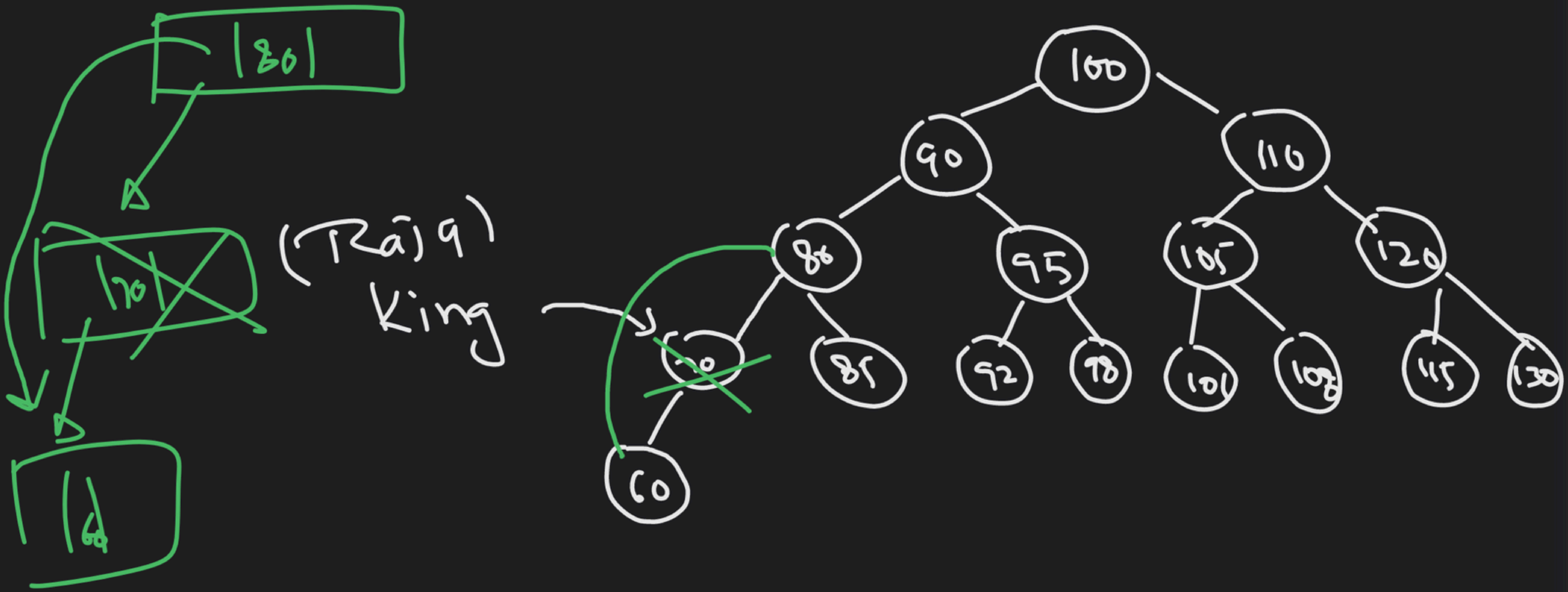


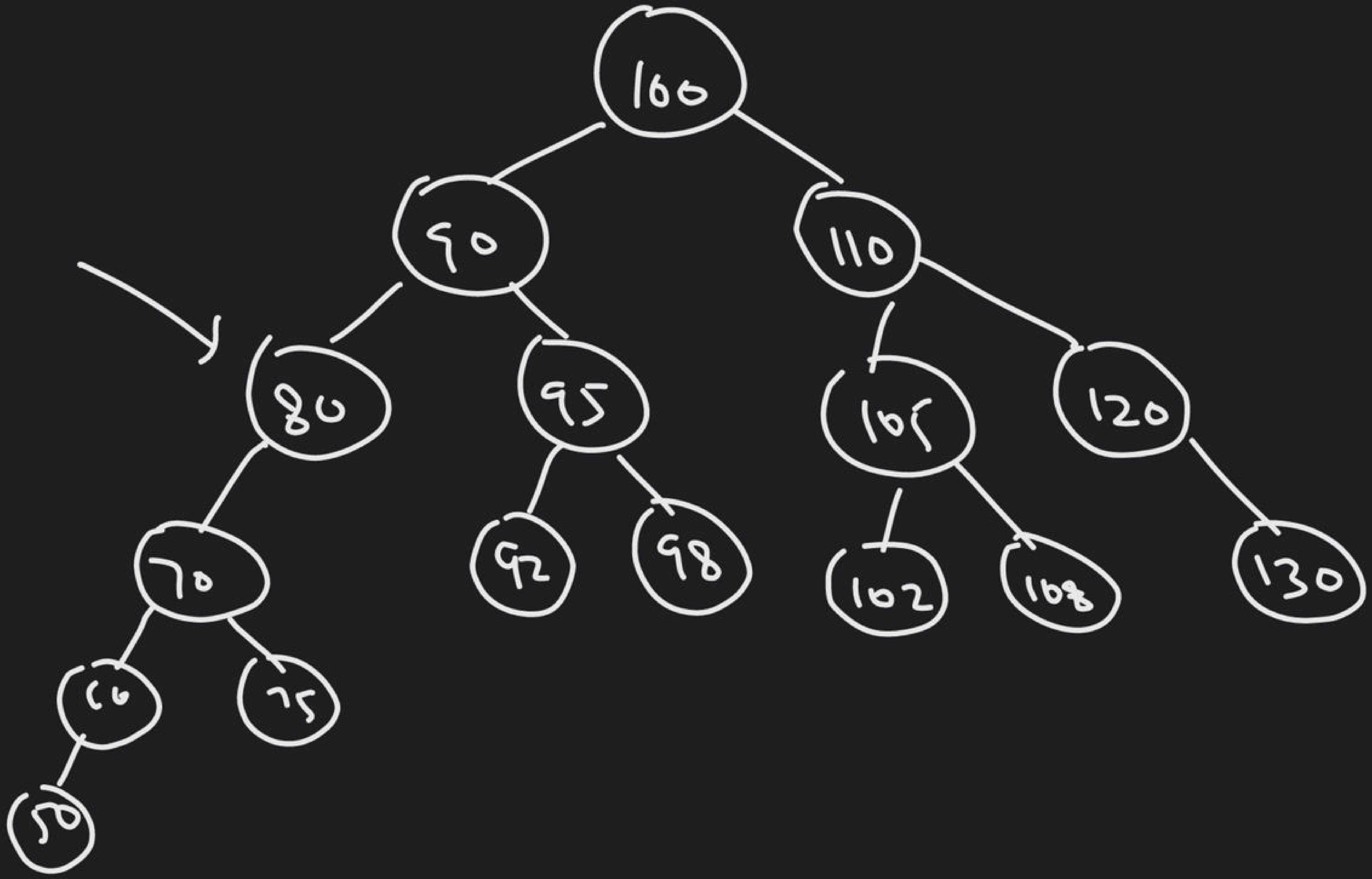


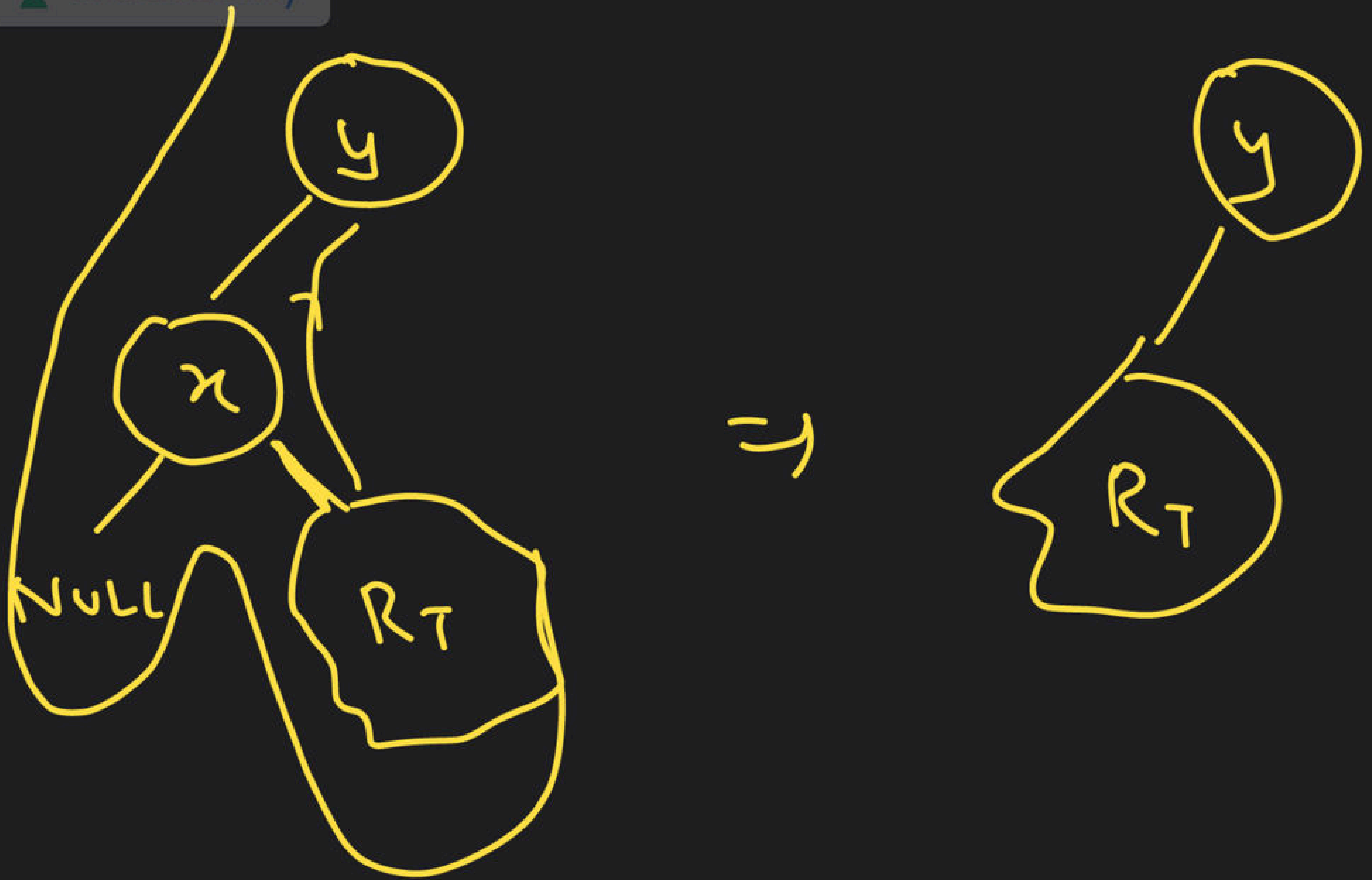
① we need to identify the parent pointer (of the node to be deleted), which is pointing to deleted node

② Set this parent pointer to NULL

Deletion of node with exaclly 1 child

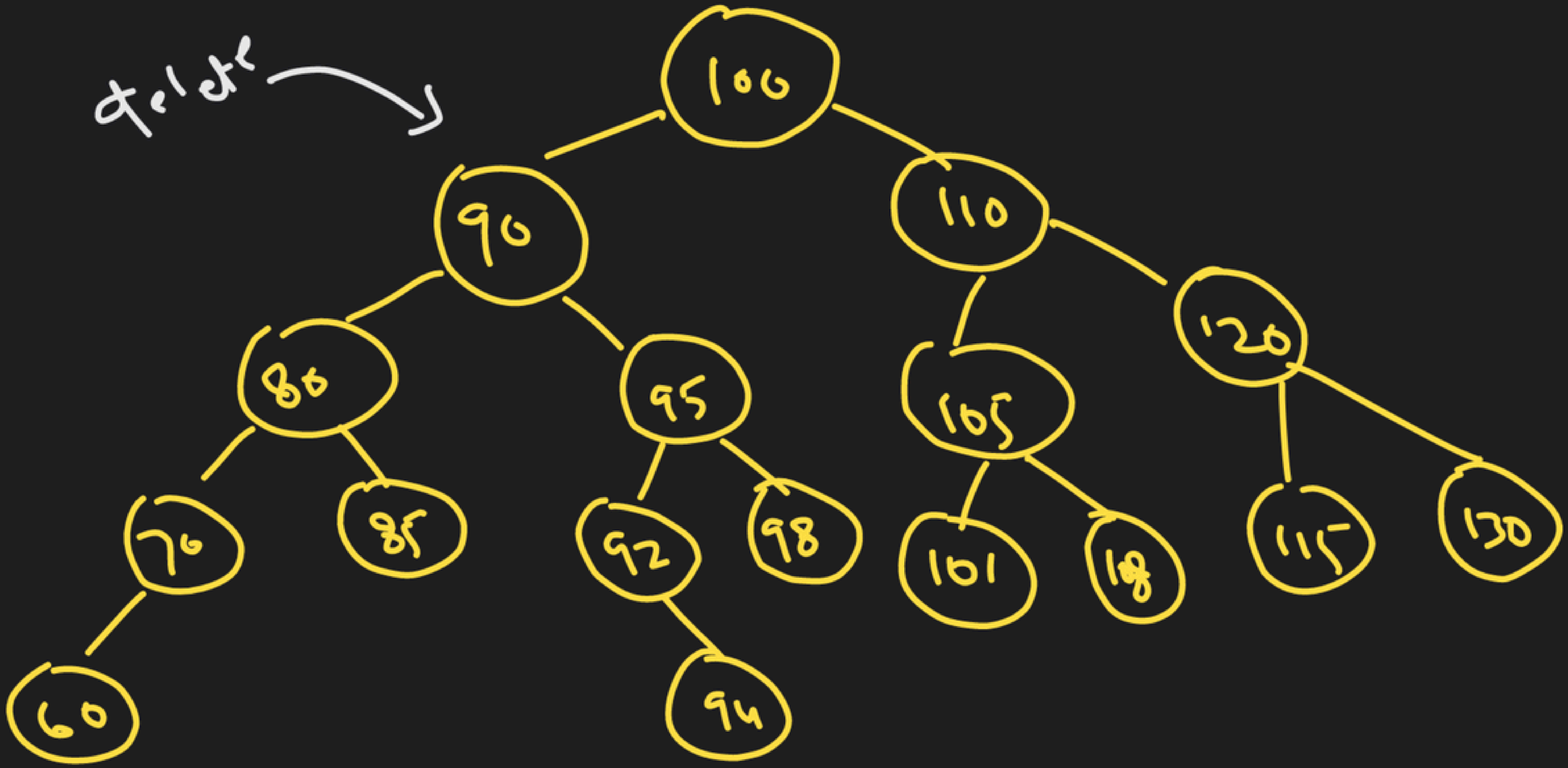


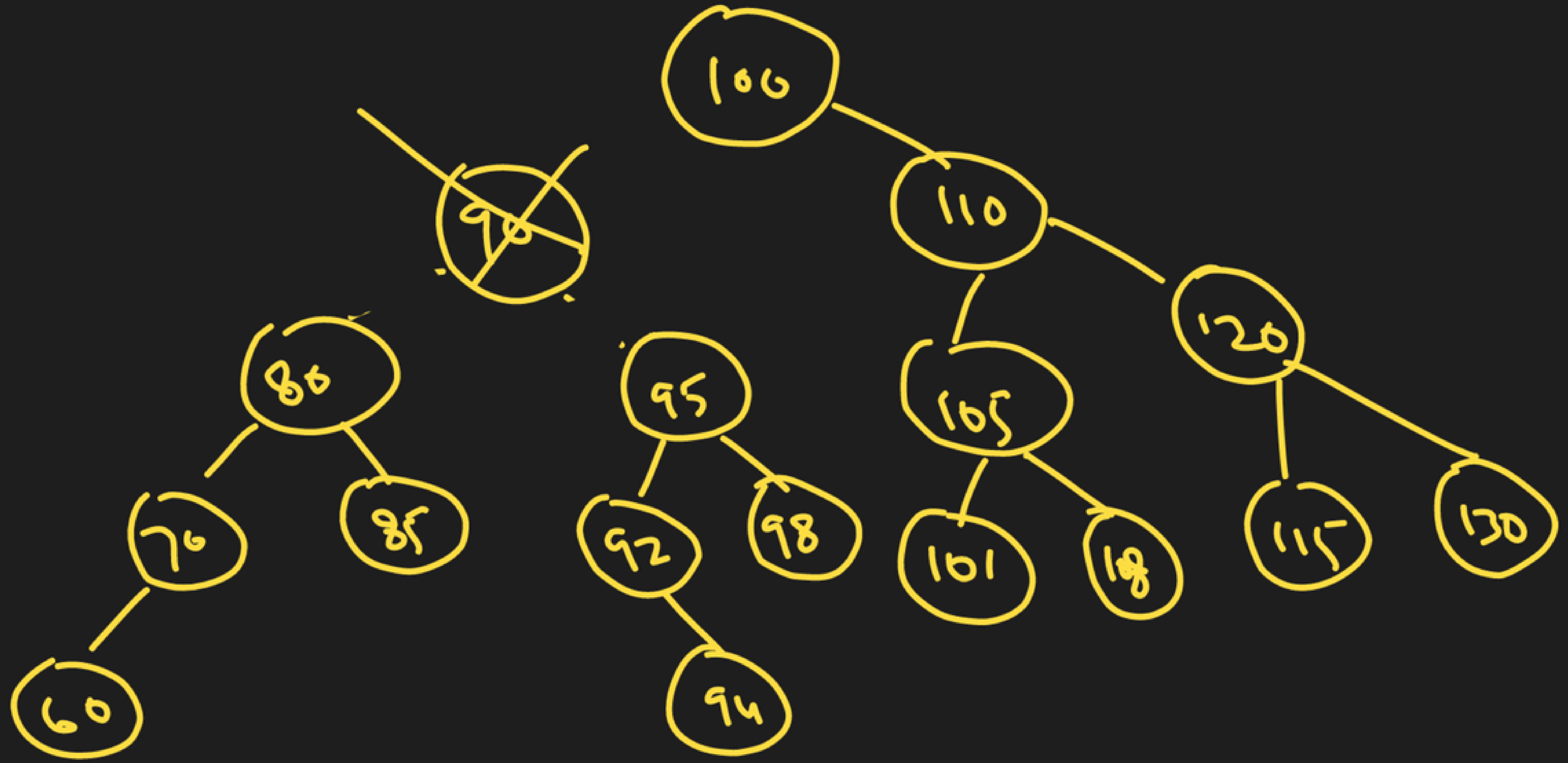

100
90
110
80
95
105
120
70
92
98
102
108
130
60
75
50



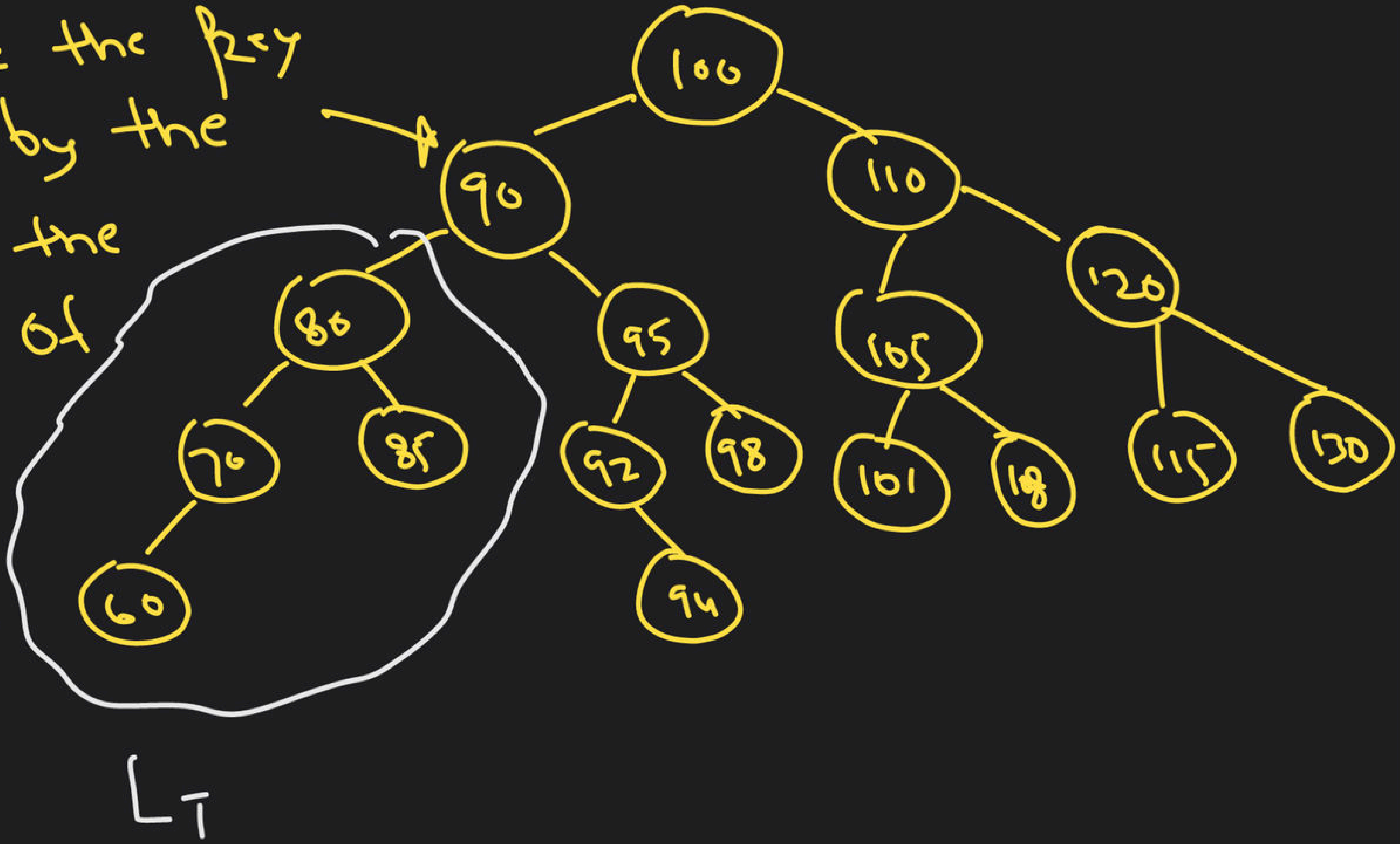
~~x~~, R_T, y

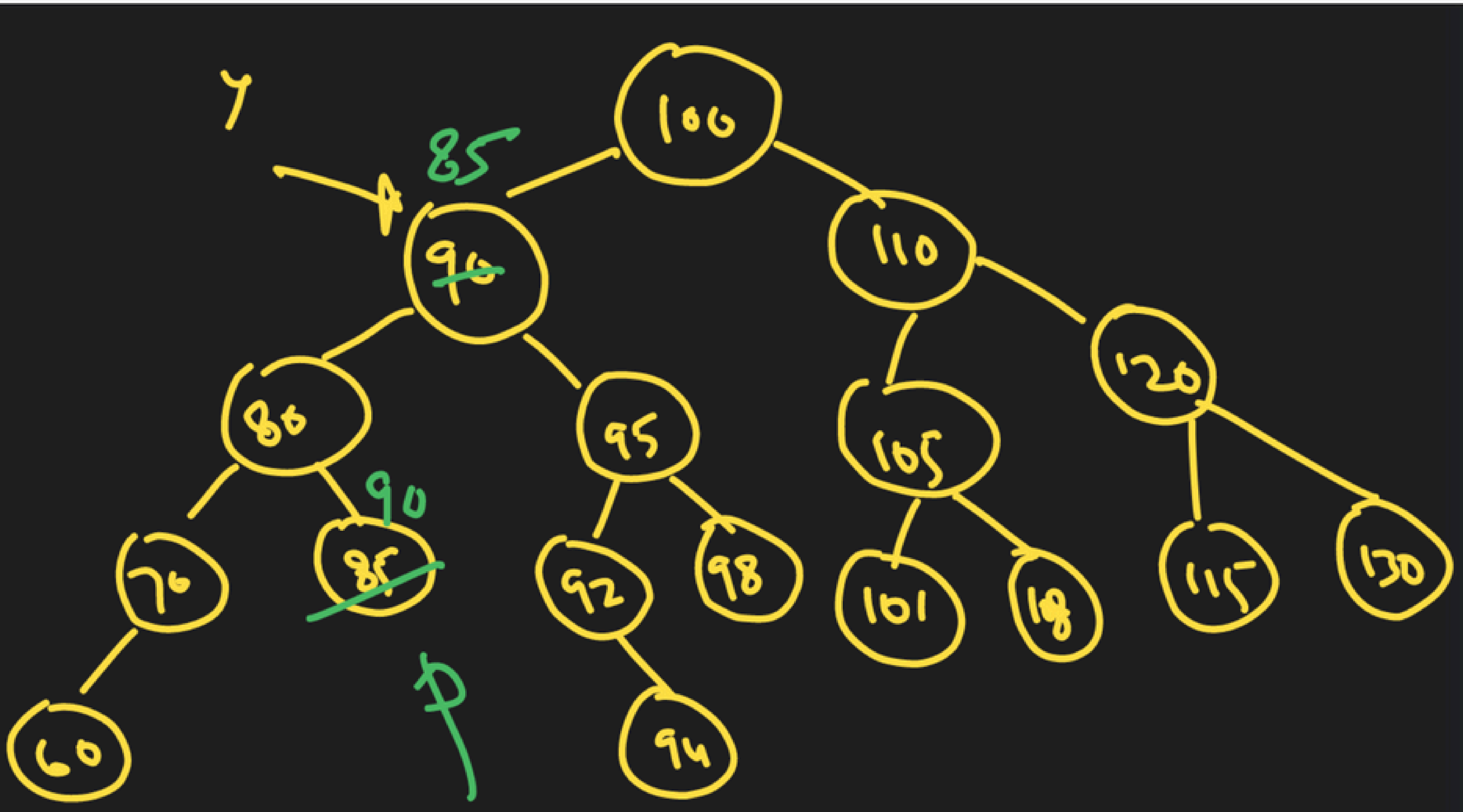
Deletion of node with 2-childs

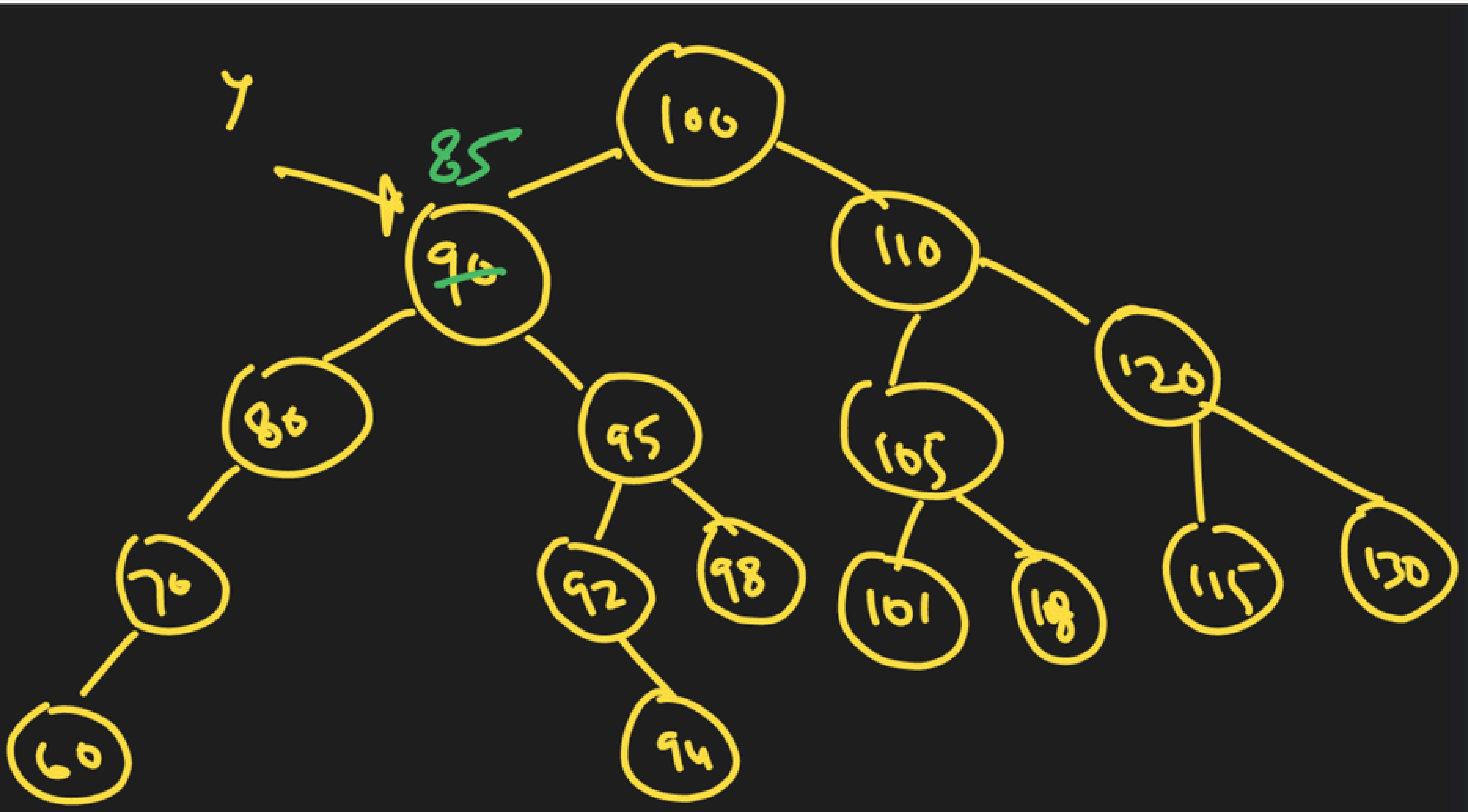



100
90
110
80
95
105
120
70
85
92
98
101
108
115
130
60
94

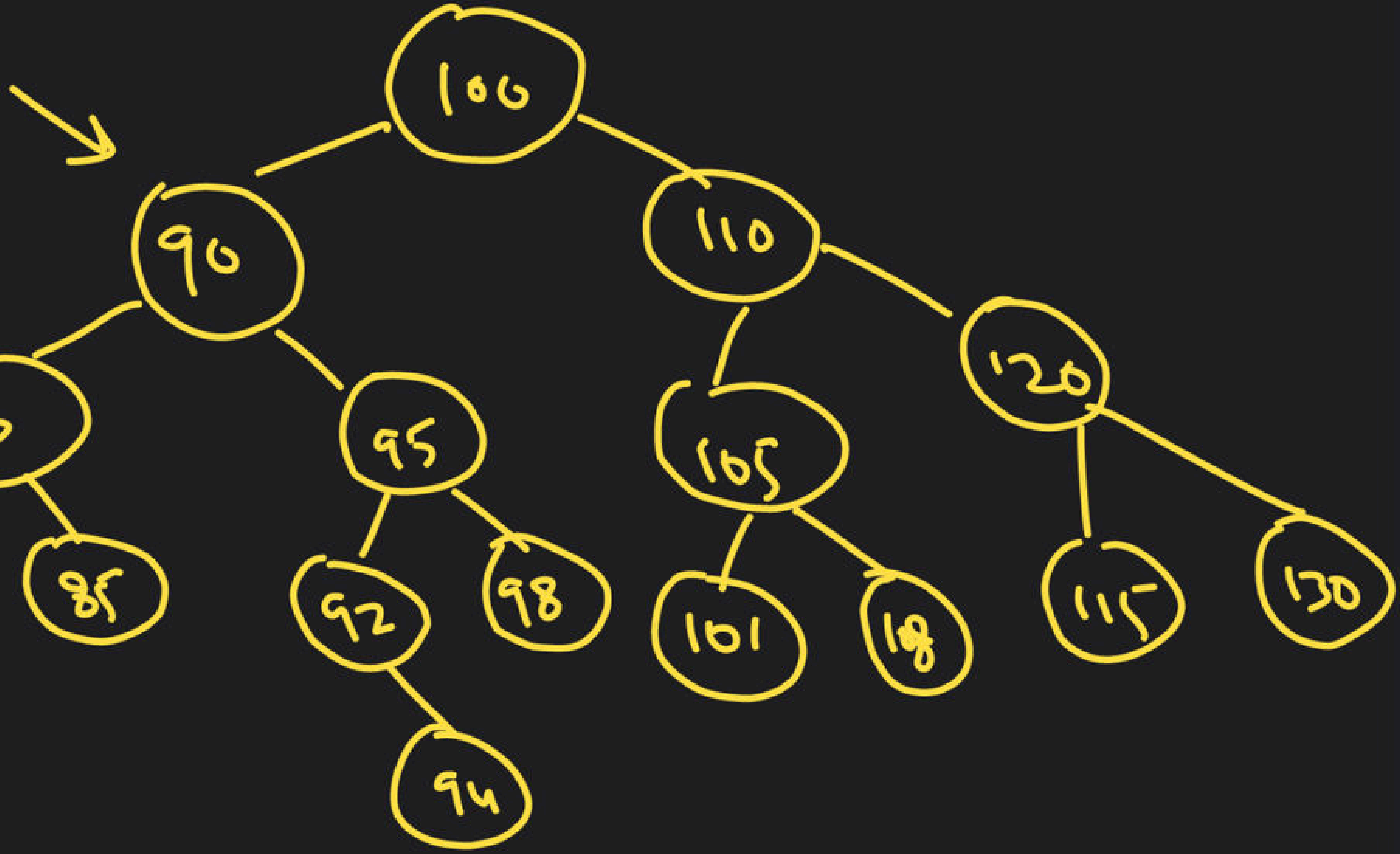
i) 1st way: Replace the key to be deleted by the largest key in the left subtree of node to be deleted & then perform deletion

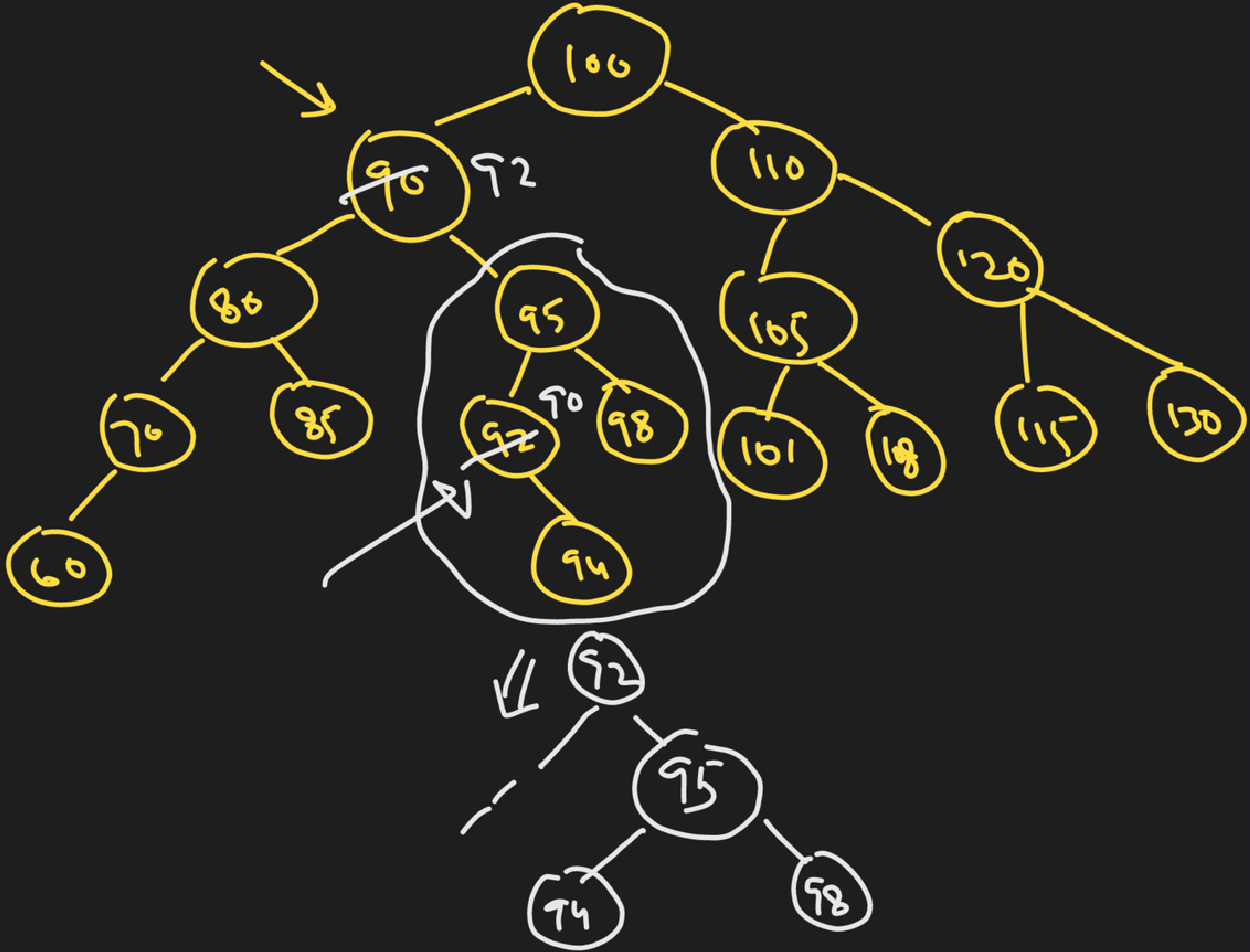



y
85
~~90~~
100
110
80
95
105
120
90
70
~~85~~
92
98
101
108
115
130
60
p
94

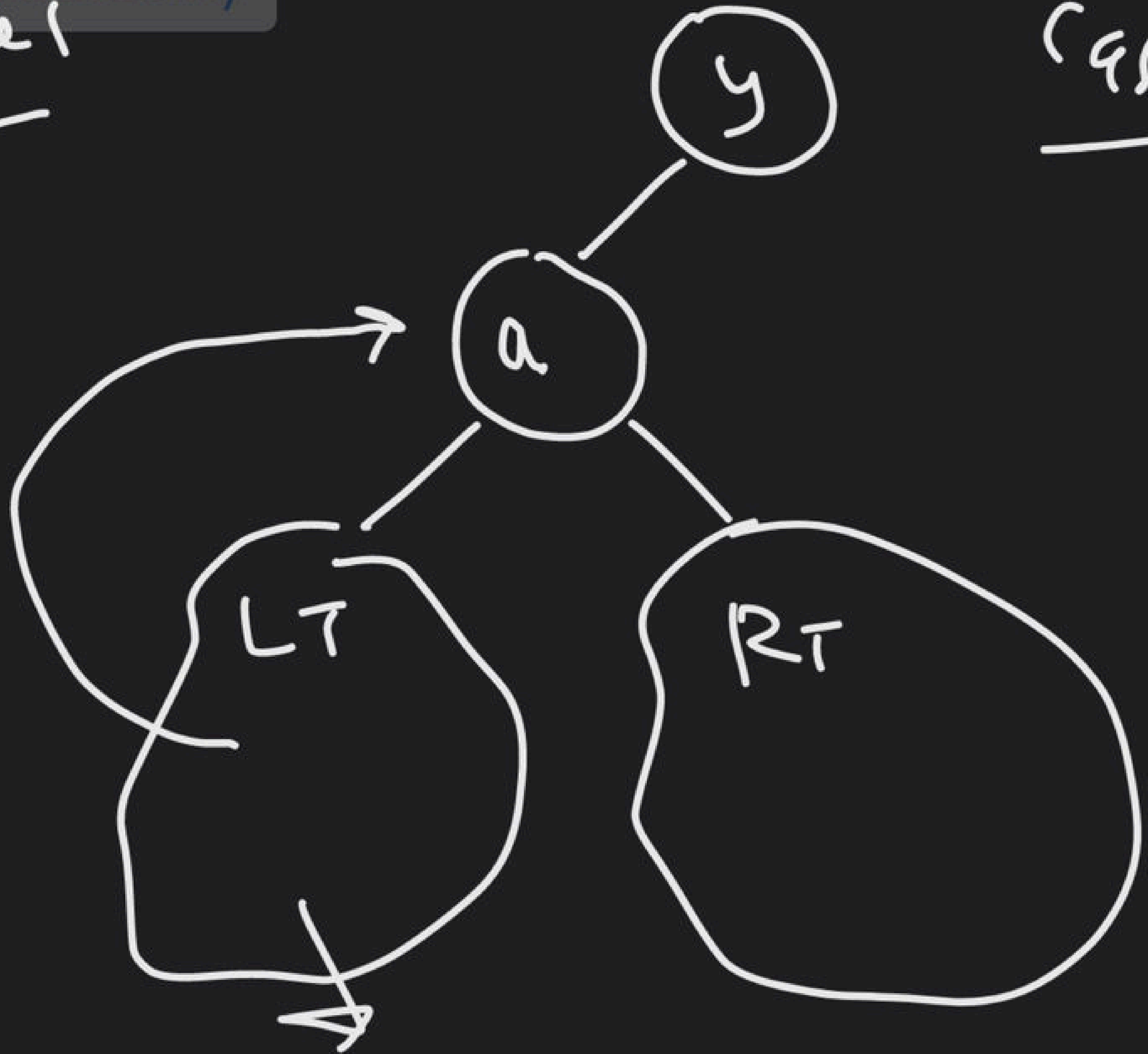

y
85
90
100
110
120
80
95
105
70
92
98
101
108
115
130
60
94

ii) Replace the key to be deleted by the smallest key in the right subtree of node to be deleted & then perform deletion



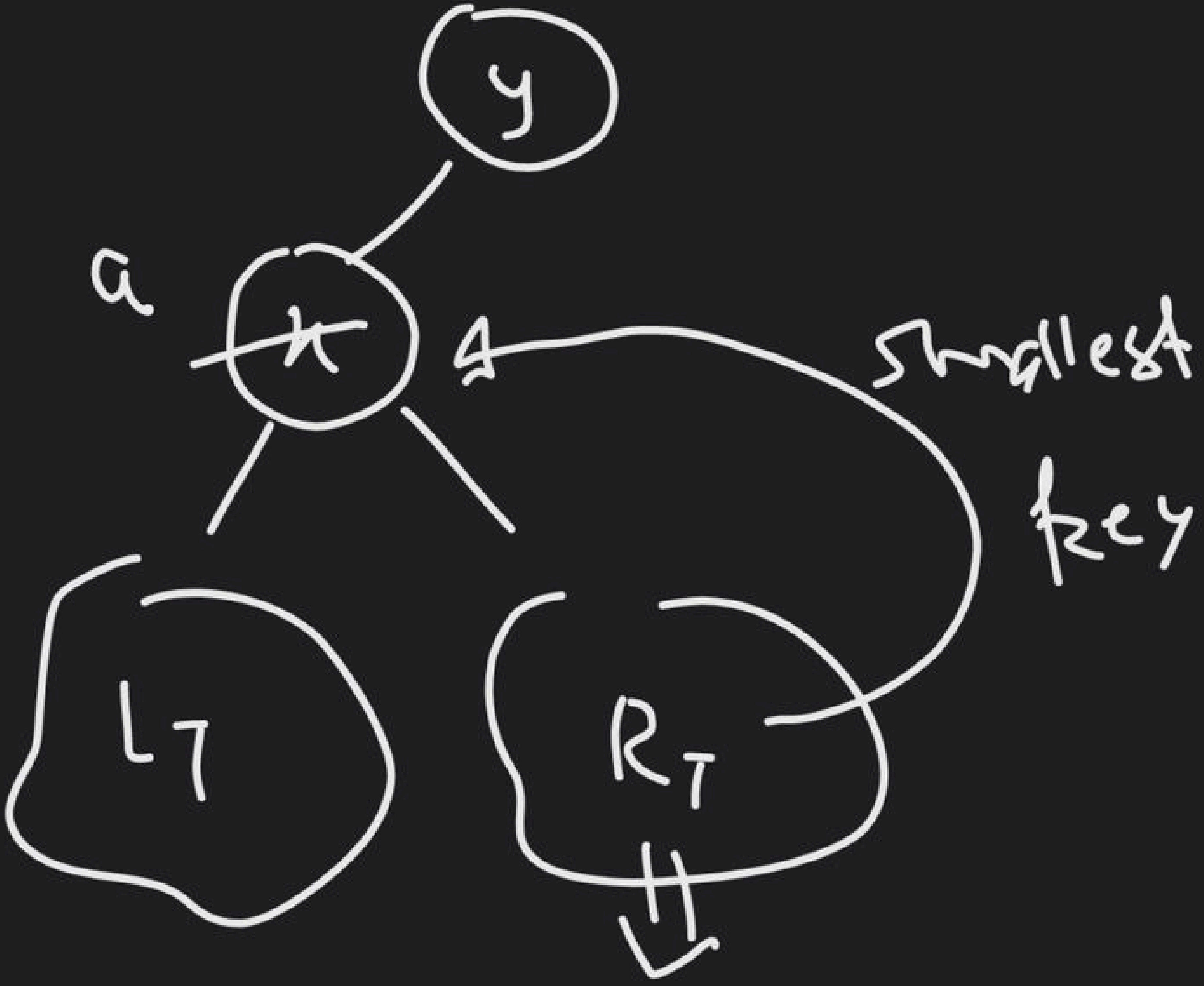

100
~~90~~
92
110
80
95
105
120
70
85
~~92~~
90
98
101
108
115
130
60
94
92
95
94
98

Case 1



Rem. keys are smaller than a

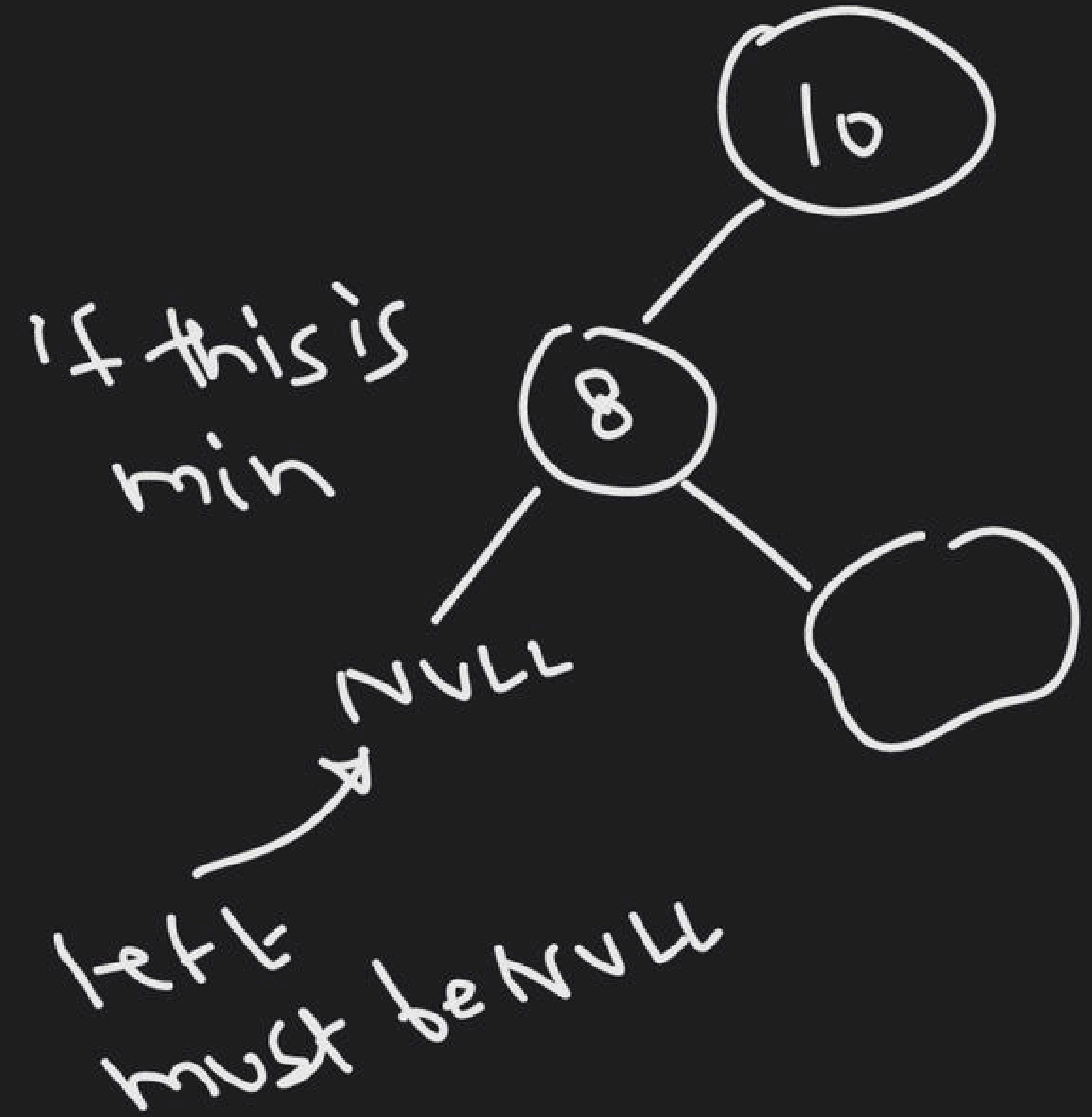
Case 2



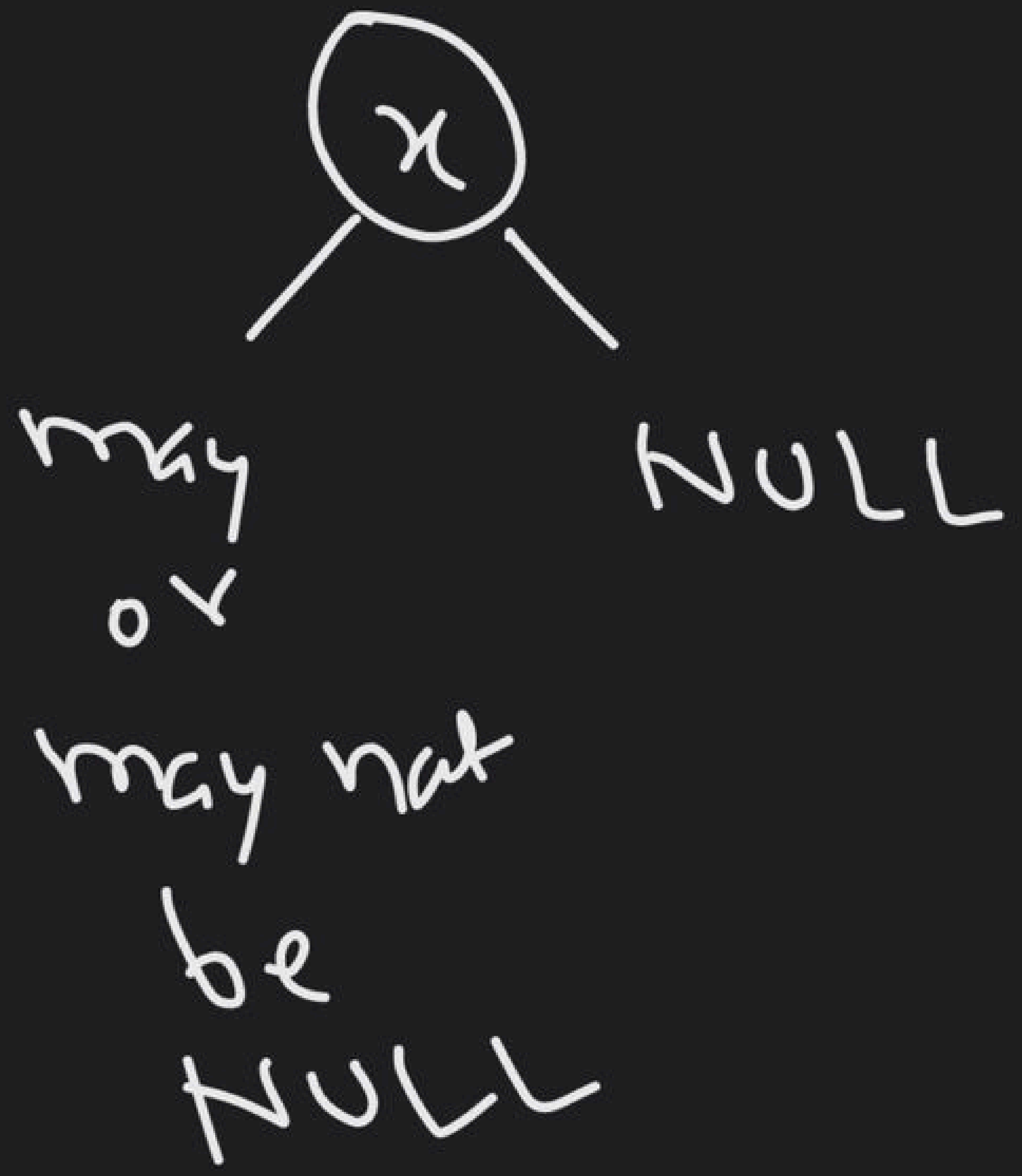
Remaining keys are greater than a

Node with min valve Can have 0 or 1 child

(No child or Only right child)



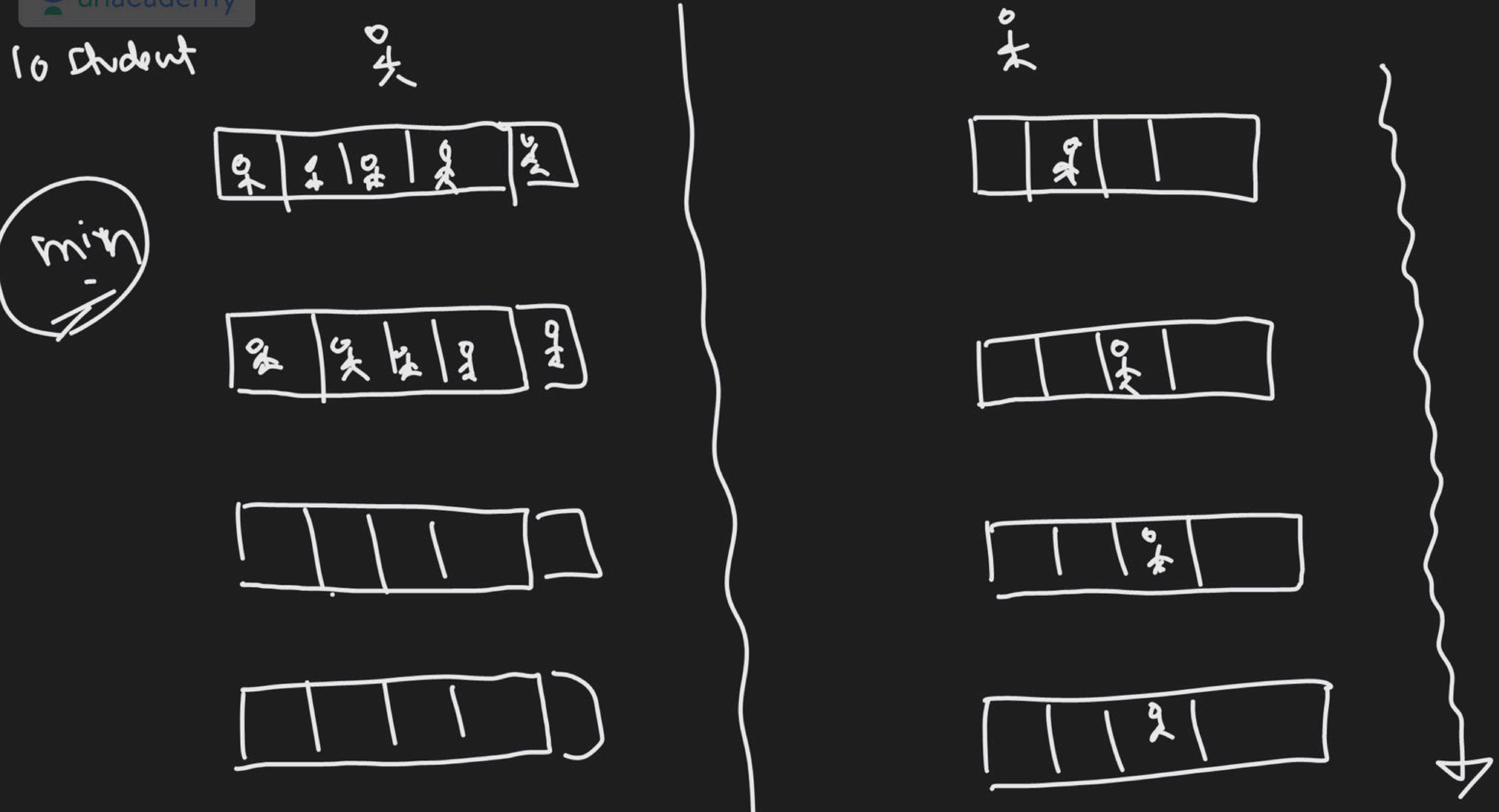
Node with max. value can have 0 or 1 child

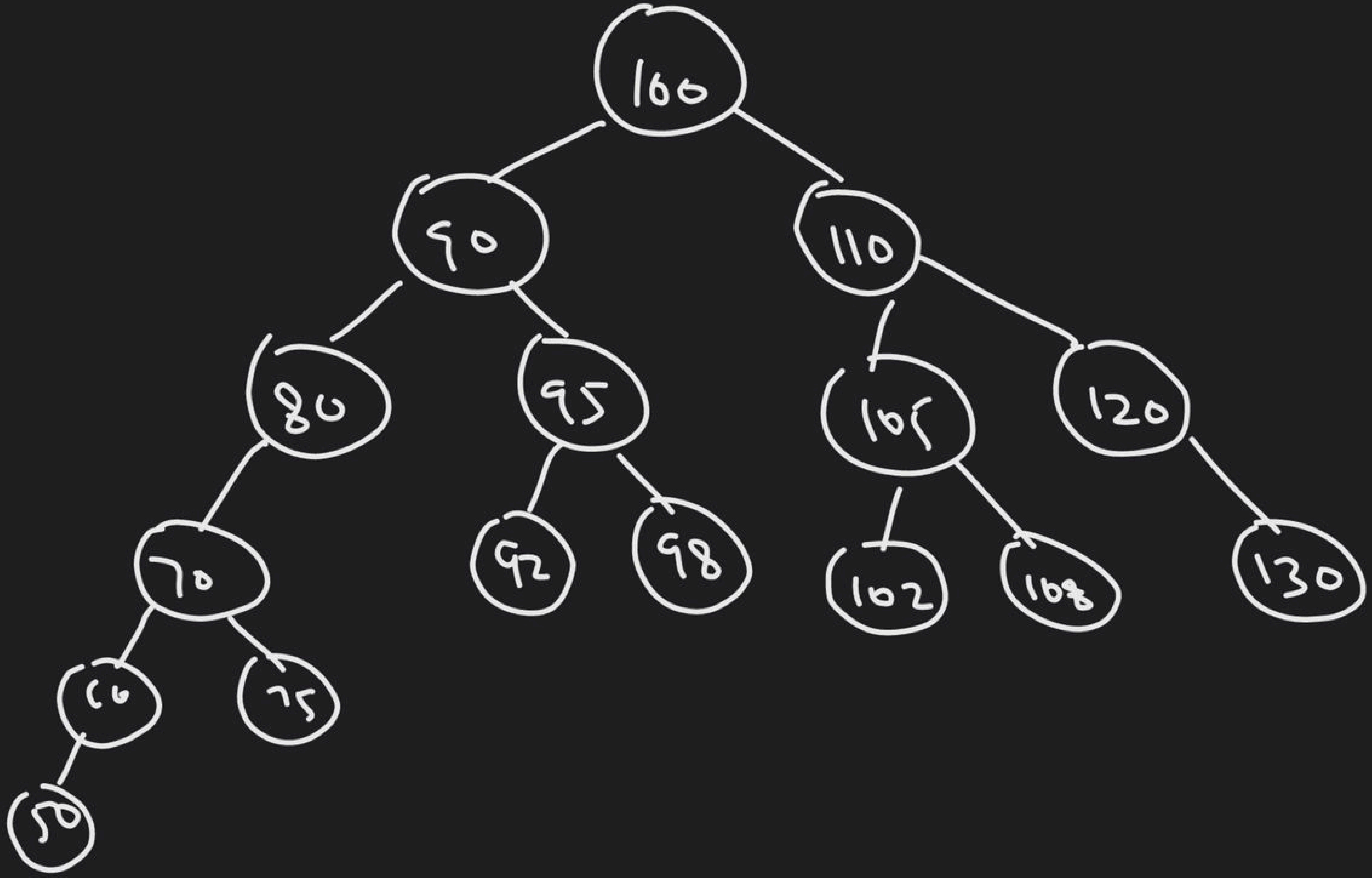


Deletion of node 2 -child → Converted into deletion of node with 0 or 1 child

BST

10 Student




100
90
110
80
95
105
120
70
92
98
102
108
130
60
75
50

Balanced BST

10:30 → Coding ✓

5:30 → C

8:00 - Data Structure

YT →

unacademy

THANK YOU!

Here’s to a cracking journey ahead!