**Question 1:**

a. First node (Key = 7):

- Begin at 1: because 7>1 → move to right (3)

- At 3: because 7>3 → move to right (8)

- At 8: because 7<8 → move to left (6)

- At 6: because 7>6 → move to right (null)

- Because the current node = null → Insert first node to this position

Second node (Key = 7):

Begin at 1: because 7>1 → move to right (3)

- At 3: because 7>3 → move to right (8)

- At 8: because 7<8 → move to left (6)

- At 6: because 7>6 → move to right (7)

- At 7: because 7=7 → Do nothing.

b. Node to delete (Key = 9):

- Begin at 1: because 9>1 → move to right (3)

- At 3: because 9>3 → move to right (8)

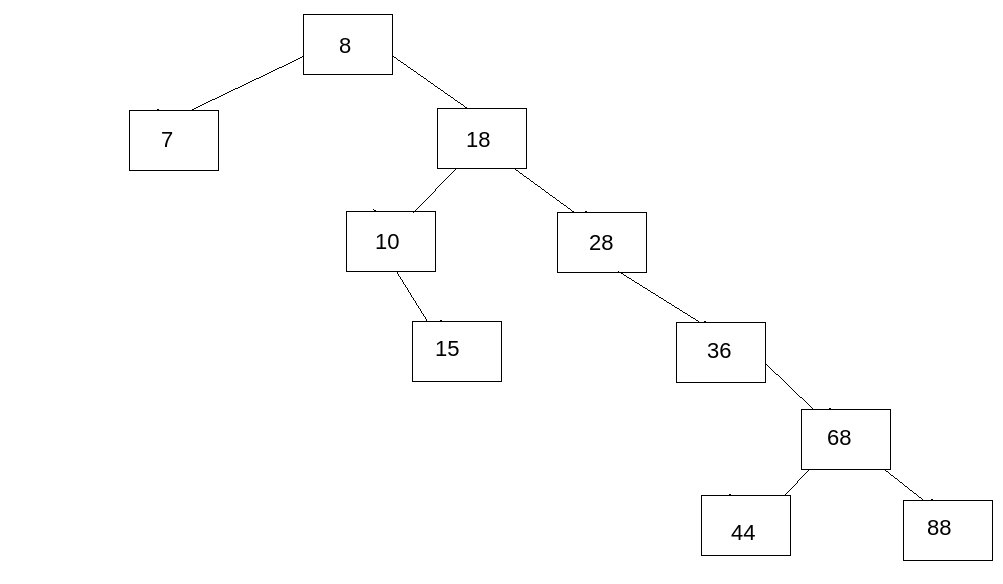
- At 8: because 9>8 → move to right (9)

- At 9: because 9=9: because left node and right node of current node = null

→ delete current node

→ change right node of node 8 to null

**Question 2:**

a.

b. Node to delete (Key = 36):

- Begin at 8: because 36>8: move to right (18)

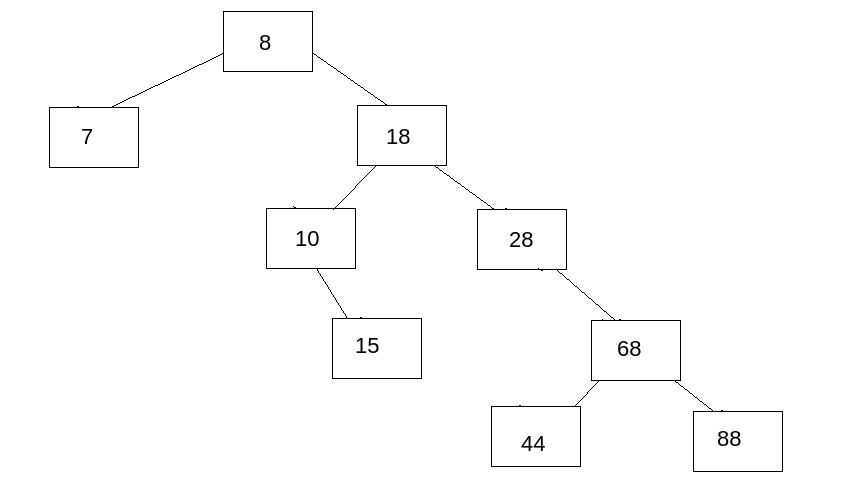
- At 18: because 36>18: move to right (28)

- At 28: because 36>28: move to right (36)

- At 36: because 36=36:

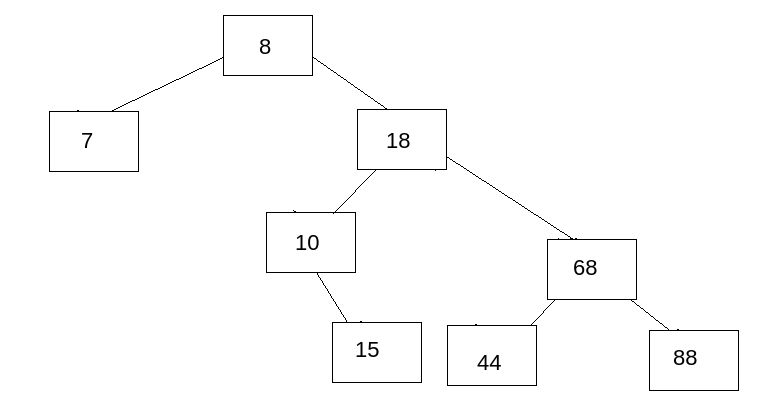
+ change right pointer of node 28 = right pointer of node 36 (point to 68)

+ delete node 36

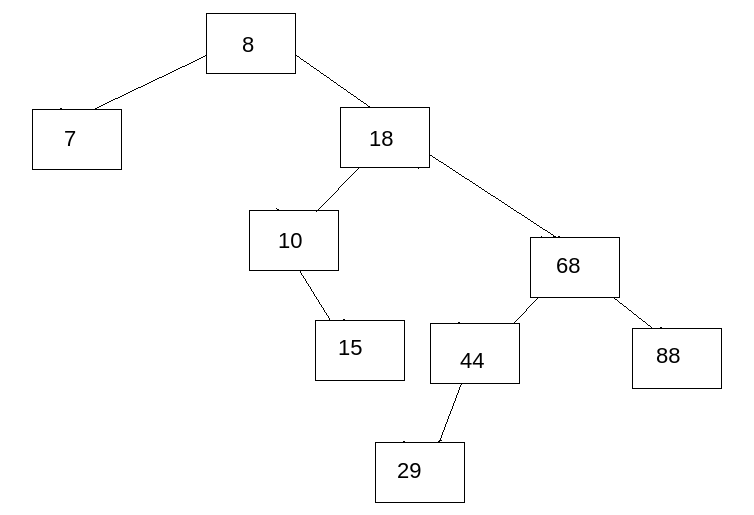


c.

- Delete node 28:



- Insert node 29:



d. - The height of current BST: 4

e.

- LNR (left – next – root):

7 → 15 → 10 → 29 → 44 → 88 → 68 → 18 → 8

- NLR (next – left – root):

88 → 29 → 44 → 68 → 15 → 10 → 18 → 7 → 8

- LRN (left – root – next):

7 → 8 → 10 → 15 → 18 → 29 → 44 → 68 → 88

**Question 3:**

1. Queue

2. Stack

3. Tree

4. Stack

5. Tree

6. Queue

7. BST

8. BST

9. Tree