Assignment - II

Name: V.D.N.S. Sai Pavan
Roll.no: 2248IA12H4
Sec : C
Sub : ADS

1. Insert the following keys into an empty B-tree of order 3 show the result step by step details 7, 8, 9, 10, 11, 16, 21, 18.

Sol. min no. of children $\lceil \frac{m}{2} \rceil = \lceil \frac{3}{2} \rceil = 2$

max. no. of child = 3

max. no of keys = $m-1 = 2$

min no. of keys = $\lceil \frac{m}{2} \rceil - 1$

$= 1$

Insert 7 :-

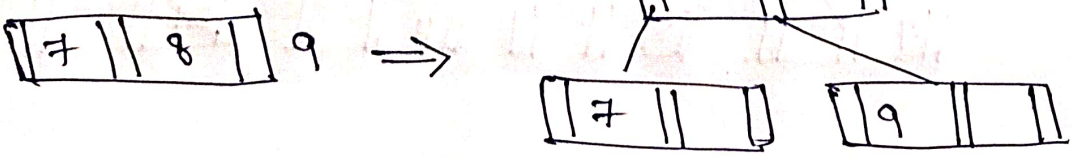
Initially B-tree is empty

Insert 8 :-

Insert the key 8 into the node maintaining sorted order 8

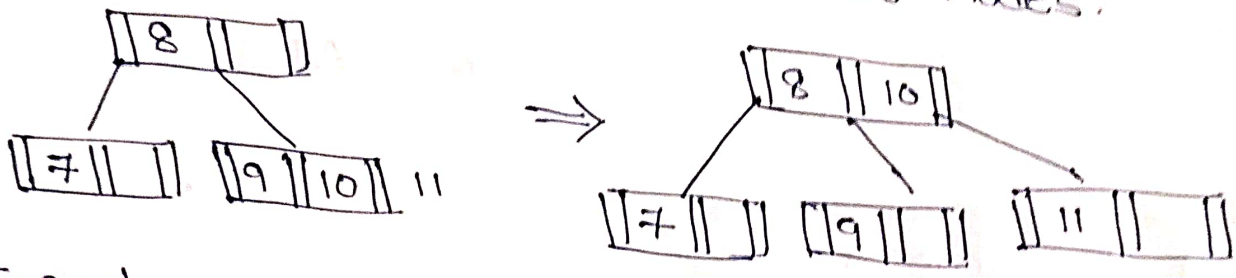
Insert-9 :-

After insert the key 9 then overflow occurs $(m-1)$, now split the node into two nodes. promote middle key to parents node Remaining b/w 2 nodes.



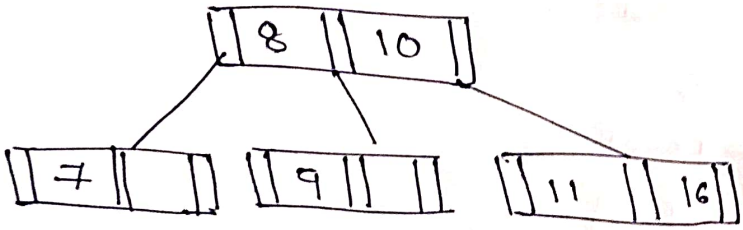
Insert '11' :-

After insert key 11, the overflow occurs now split node into two nodes, promote middle key into ~~not~~ node remain between two nodes.



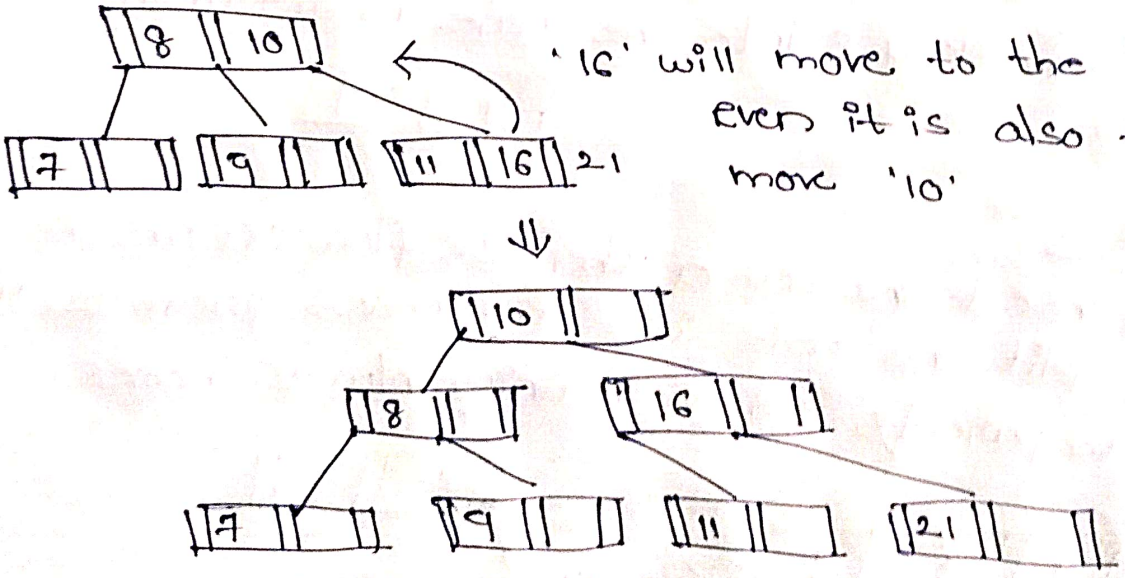
Insert '16' :-

Insert the key 16 into node Here 16 is ~~gre~~ greater than 11. Then place to it's right side.



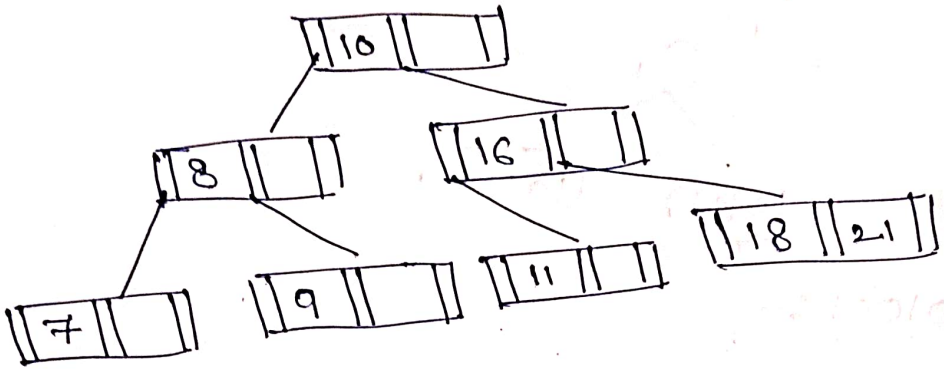
Insert '21' :-

After inserting the key '21' there overflow condition is violating. Now, split the node into two nodes.



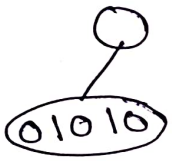
Insert 18 :-

Insert the key '18' into node. Here 18 is greater than 16 and less than 21. so place 18 on leftside of 21.



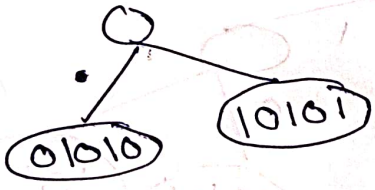
2.) Construct the trie for the keys 01010, 10101, 10110, 01001, 00000, 11011, 00110, then delete key 00000, 10101 and 11011. How does the deletion process works in binary trie.

Sol. Insert 01010 :-

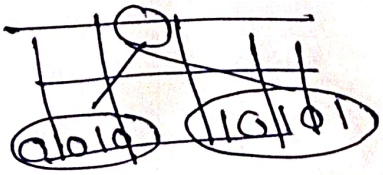


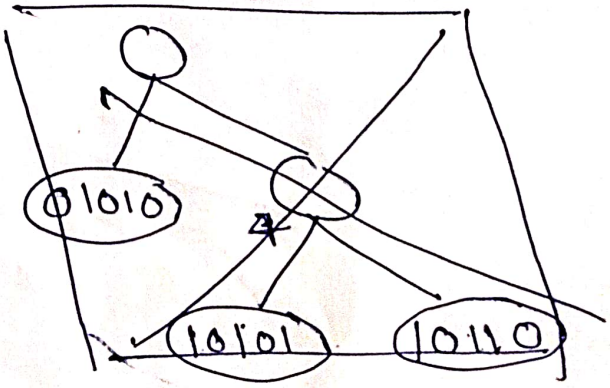
Here the trie is empty and the first bit is '0' so insert it in the left sub-tree

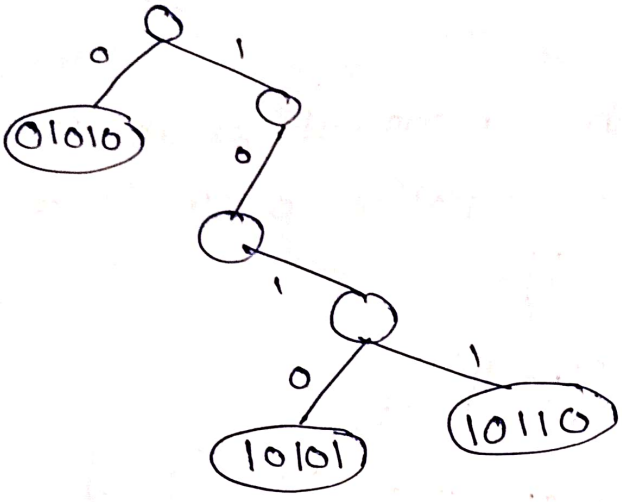
Insert '10101' :-



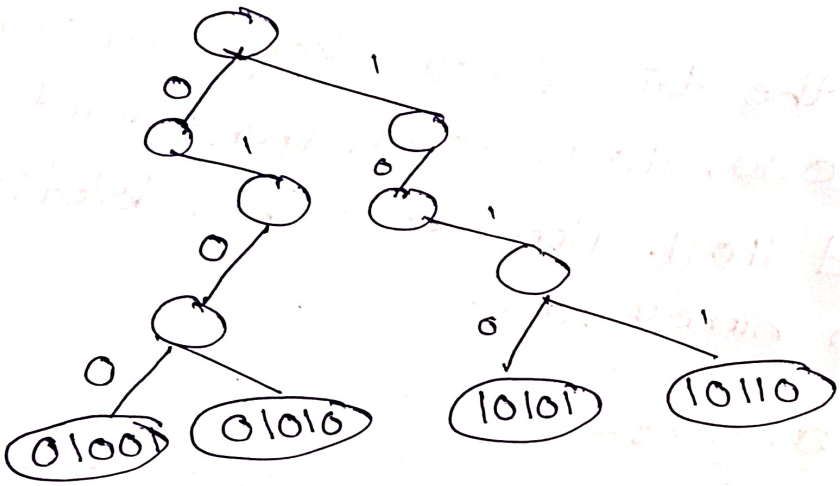
Insert 10110 :-



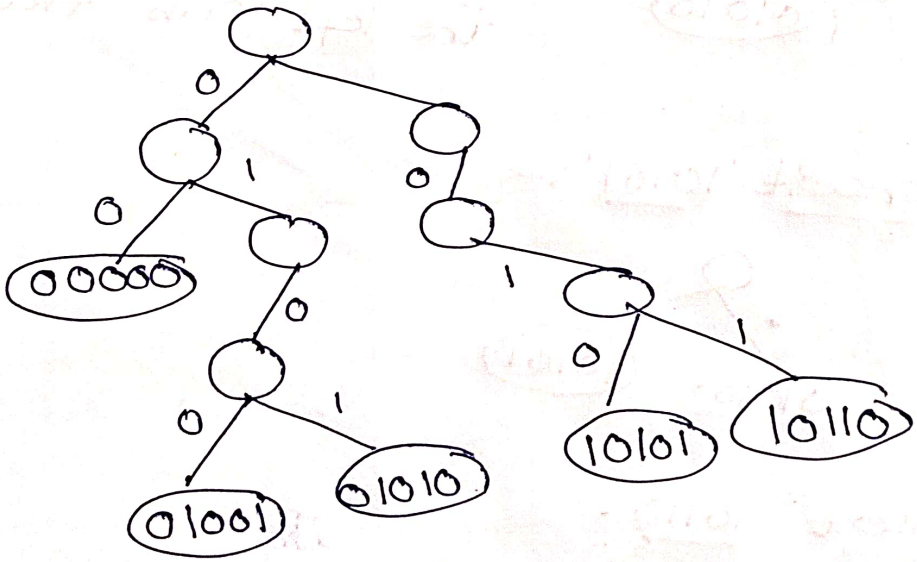




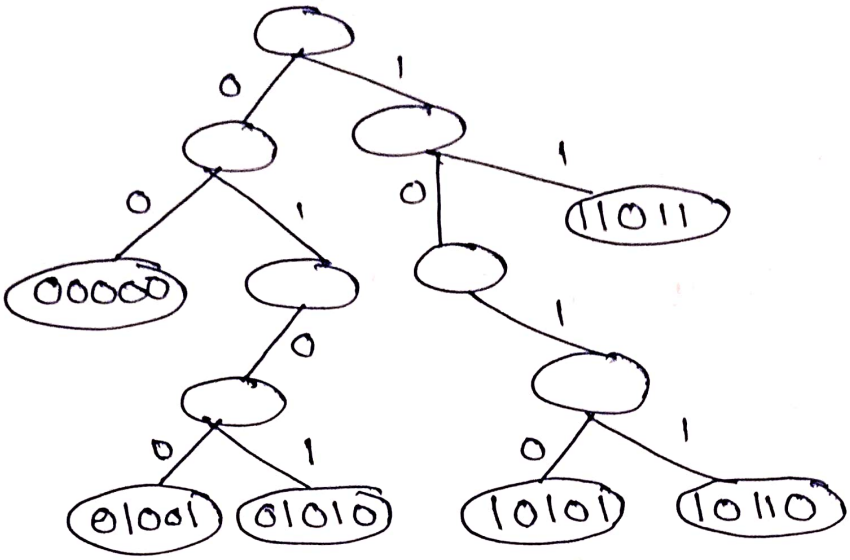
Insert 01001 :-



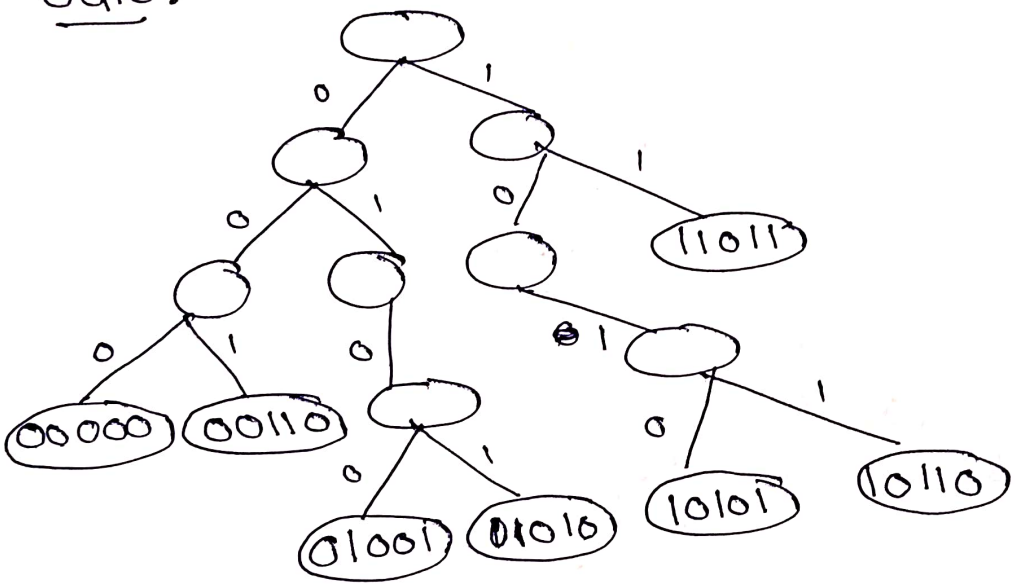
Insert 00000 :-



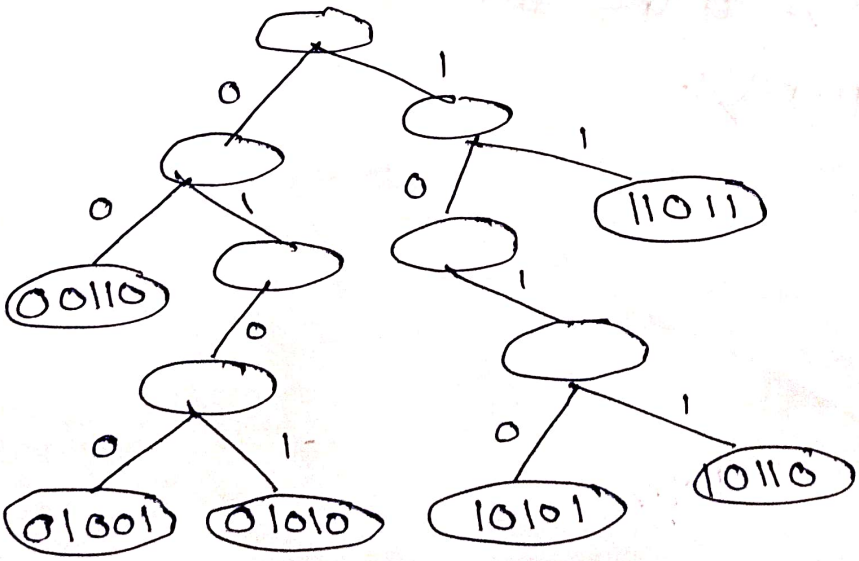
Insert 11011 :-



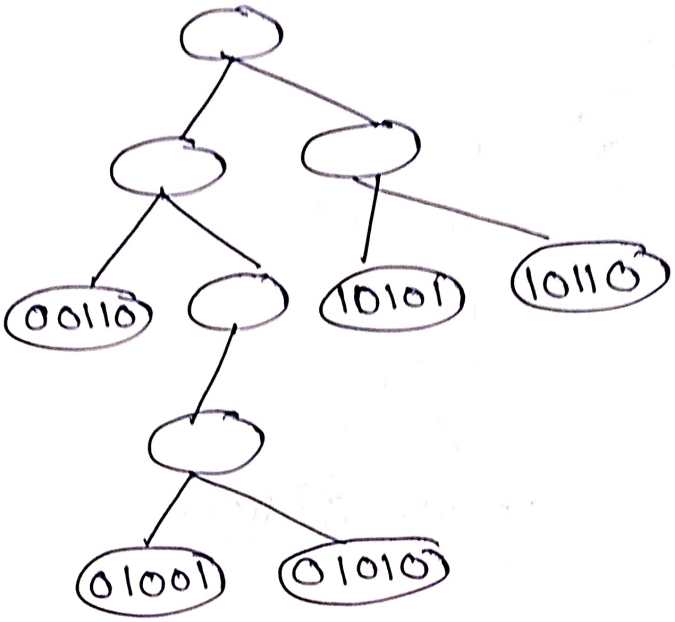
Insert 00110:



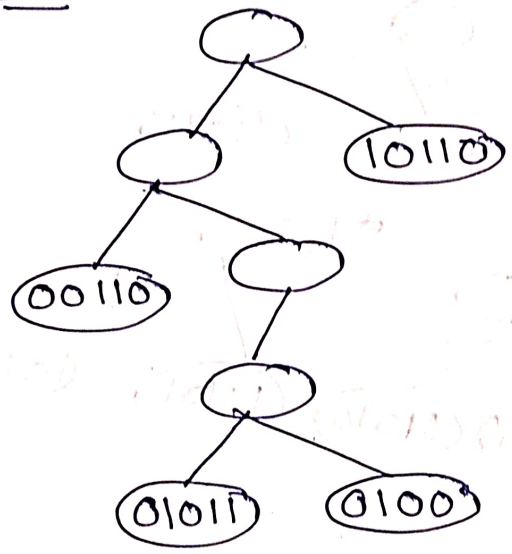
Delete '00000' :-



Delete '10101' :-



Delete '11011' :-



this is the final tric after deletion of elements.