11/8/21

MULTI-WAY SEARCH TREE

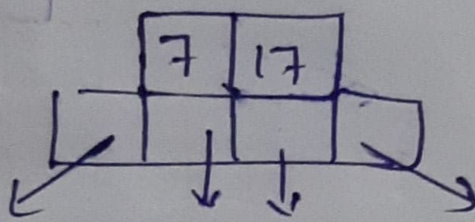
→ Generalization of Binary Search Trees.

→ Number of keys stored per node

— 1 (BST)

— >1 (Multiway Search tree)

Order – Num of maximum possible children.



order - 4

→ A m-way search tree T may be an empty tree. If T is non-empty it satisfies the following properties –

(a) Each node has at most m children.

(b) If a node has k child nodes ($k \le m$), then the node has exactly $(k-1)$ keys.

(c) The keys in each node are in ascending order.

(d) The keys in the left subtree of a key 'i' are smaller than 'i'.

(e) The keys in the right subtree of a key 'i' are greater than 'i'.

eg:-

Operations - ① Insertion ② Search ③ Deletion.

INSERTION -

Case-1 -

If a node can accomodate k, insert k in to the node and adjust the pointers.

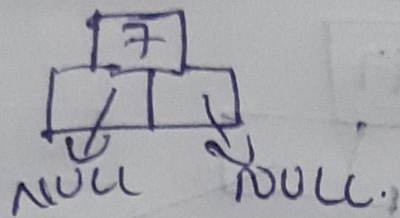
Case-2 -

If a node can not accomodate k, insert k in to a new node at the next level.

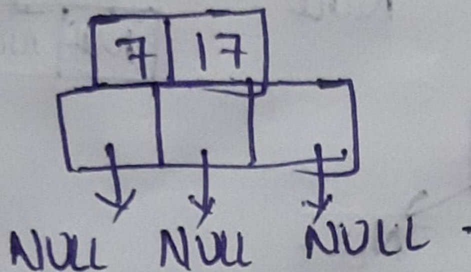
egr

insert 7

Order - 3

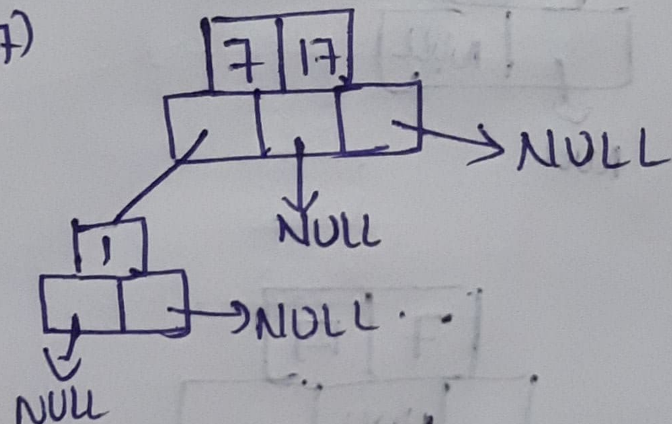


insert 17 -

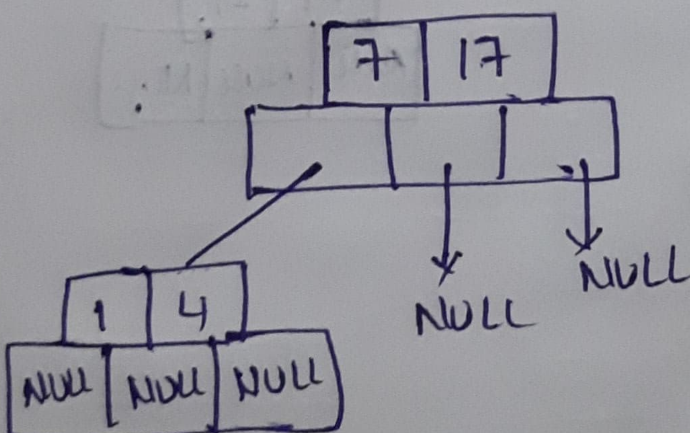


(17 > 7)

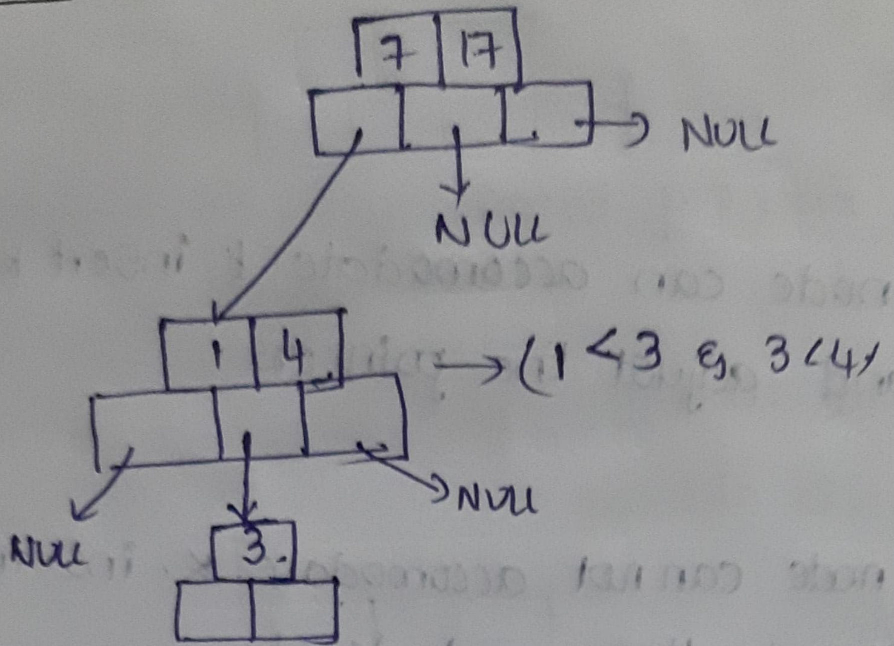
insert 1 - (1 < 7)



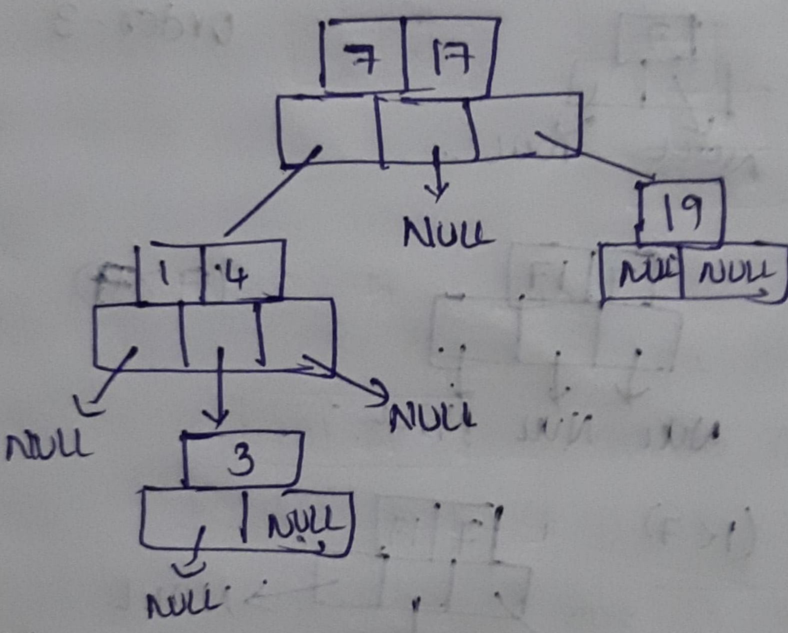
insert - 4 - (4 < 7) & (1 < 4)



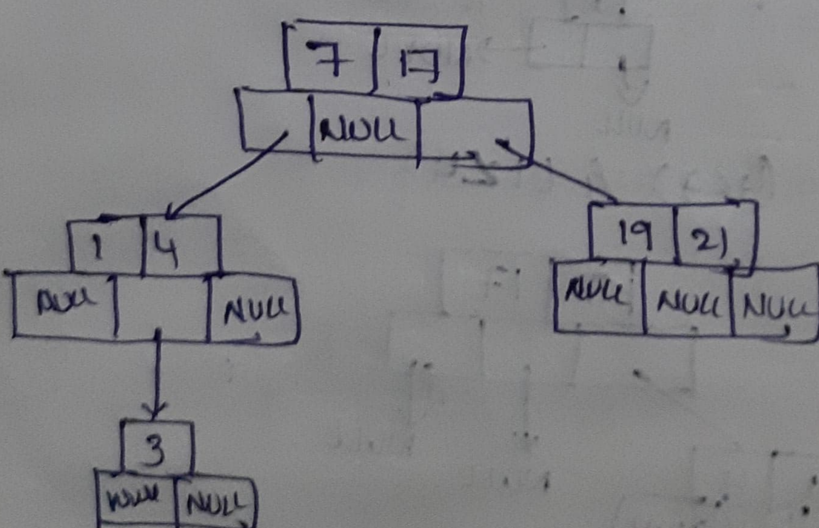
insert -3 (3<7).



insert -19 19>7 & 17<19.



insert -21



Search-3 $3<7$ - move left

$3>1$ - ~~move~~ & $3<4$ - in b/w there 2.

search-5 $5<7$ - move left

$5>1$ & $5<4$ - move right

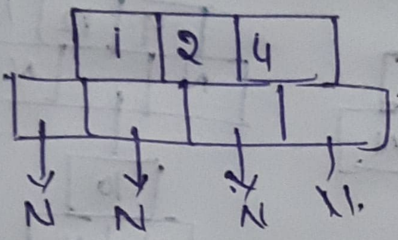
But right subtree is empty

So, search is unsuccessful.

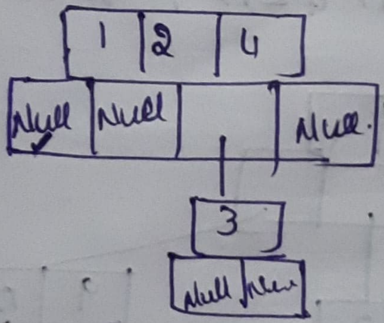
16/6/21

Q: ORDER = 4.

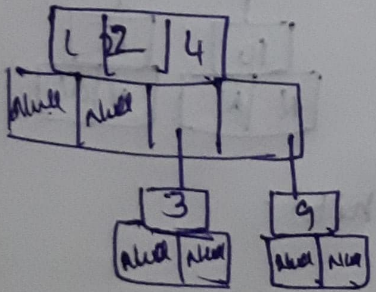
Insert 1, 2, 4, 3, 9, 12, 16, 10, 15.



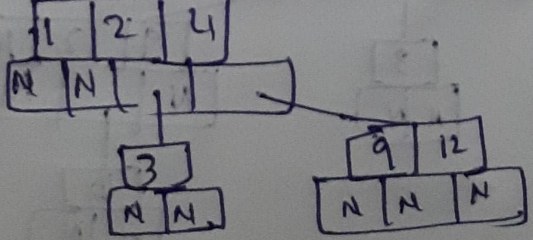
Insert -3 $1<3$; $2 \not< 3$; $4>3$.



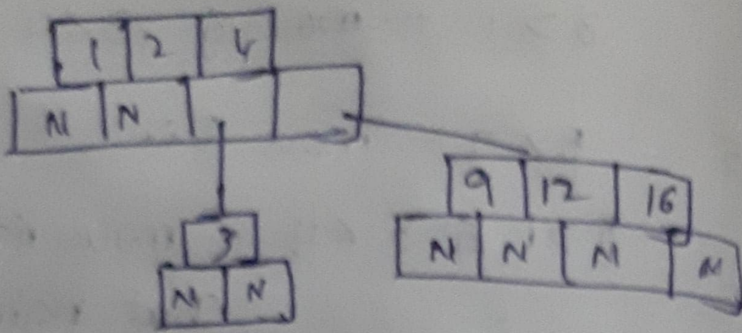
Insert -9 $1<9$; $2<9$; $4<9$.



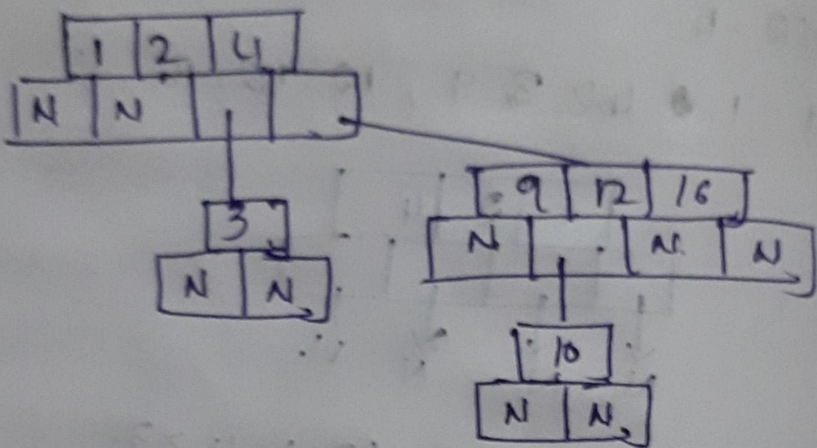
Insert -12 $1<12$; $2<12$; $4<12$; $9<12$.



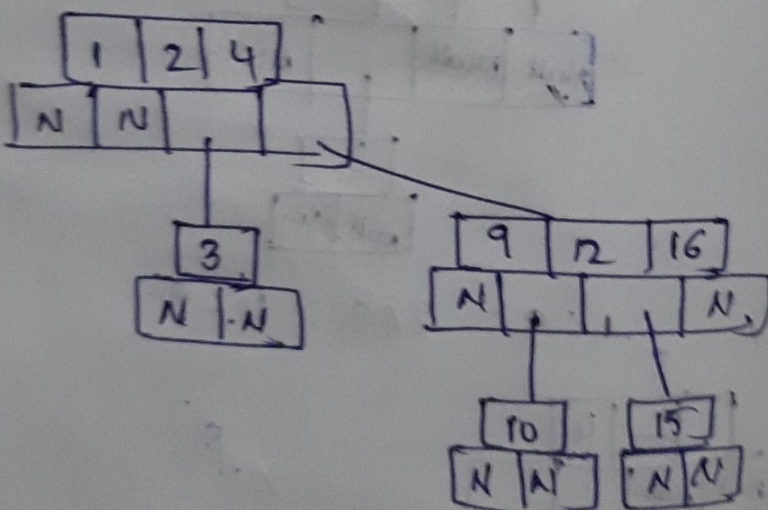
insert-16 1<16 ; 2<16 ; 4<16 ; 9<16 ; 12<16.



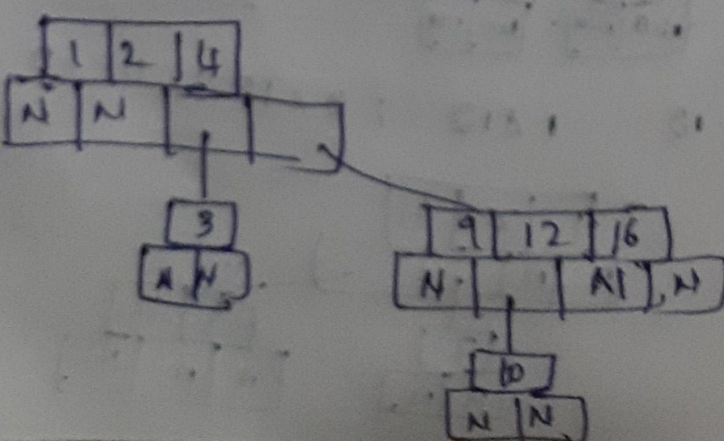
insert-10 1<10; 2<10; 4<10; 9<10; 12>10



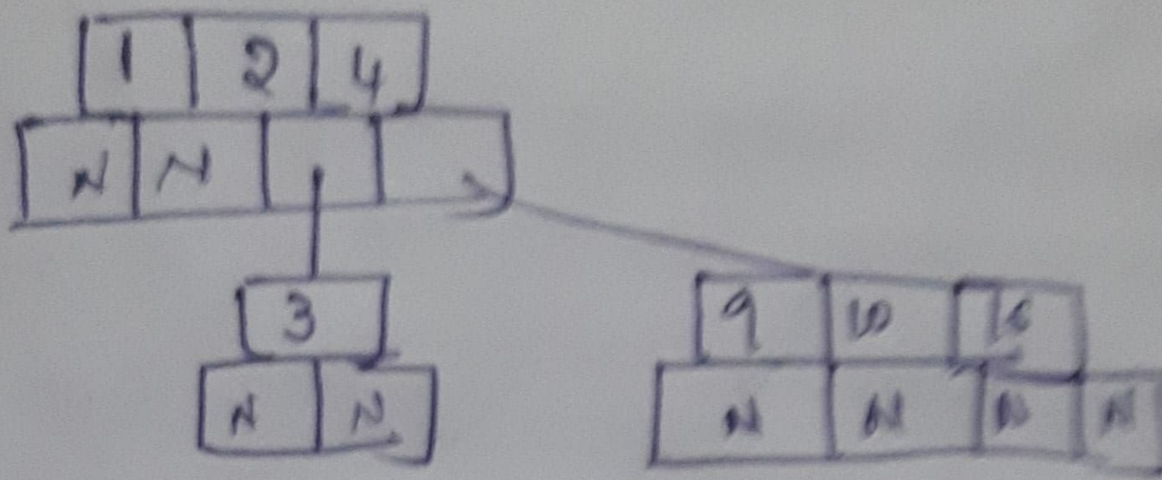
insert-15. 1<15; 2<15; 4<15; 9<15; 12<15; 16>



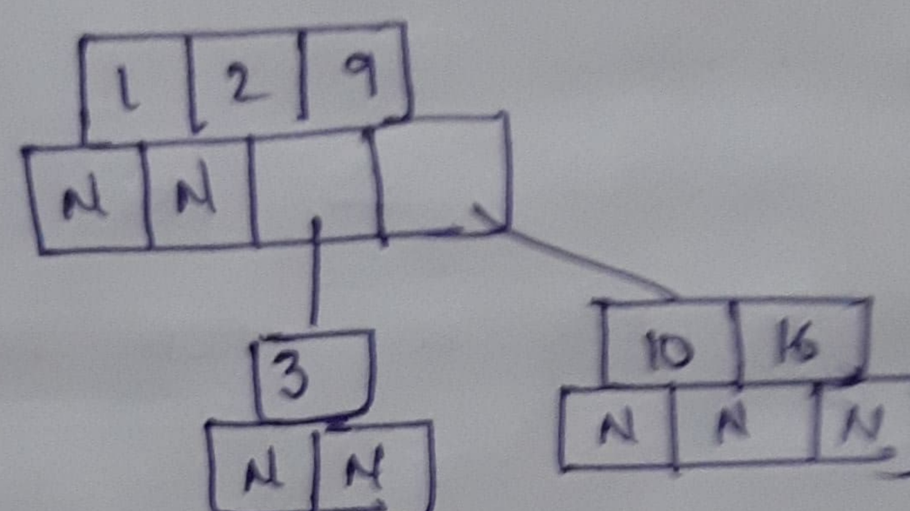
Delete-15 - No Rchild & No Lchild.



Delete-12 - 12 has left child.



Delete-4 4 has both L child & R child.



ANALYSIS OF M-WAY SEARCH TREES

→ Complexity of any operation is $O(h)$.

→ Minimum number of elements 'h'.

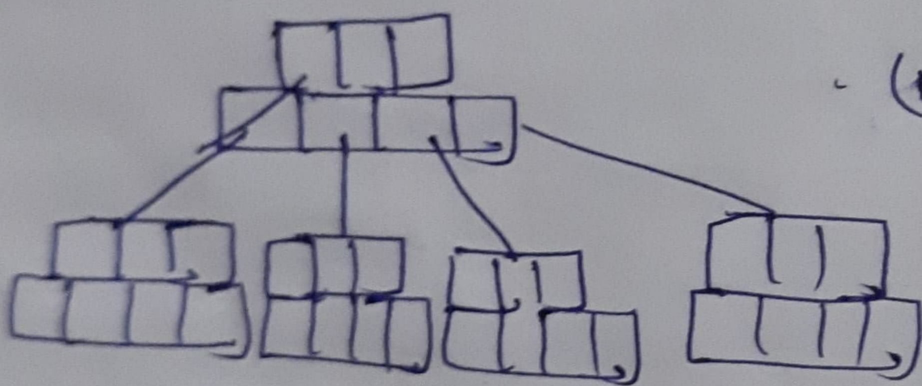
→ Maximum number of elements -

order → No of children $= m$

No of keys $= m-1$

height - 1

height - 2



- - -

At height h -

$m^{(h-1)}(m-1)$ keys.

Total number of elements =

$$= (m-1) + m(m-1) + m^2(m-1) + \cdots + m^{h-1}(m-1)$$

$$= (m-1)(1 + m + m^2 + \cdots + m^{h-1})$$

$$= (m-1)\frac{(m^h - 1)}{(m-1)}$$

$$= m^h - 1$$

Total number of elements $= m^h - 1$

→ If a M-way search tree has n elements, the height h varies from a minimum of n to a maximum of m^{h-1}.

→ Best case -

$$n = m^h - 1$$

$$\log_m n = (h-1)\log_m m \qquad \log_m n = h\log_m m - \log_m m$$

$$\log_m n = h-1 \qquad m^h = n+1$$

$$h\log_m m = \log_m n + \log_m 1$$

$$\boxed{h = \log_m n + 1} \qquad \boxed{h = \log_m (n+1)}$$

→ Worst case - $O(n)$