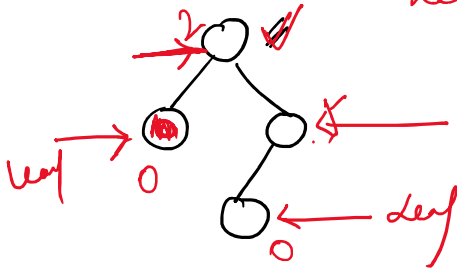


## Heap Sort -

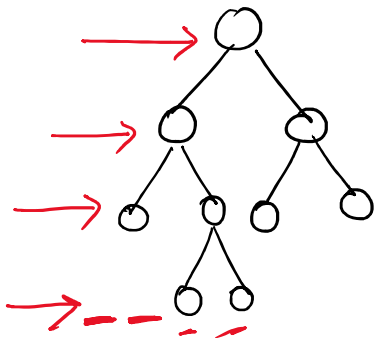
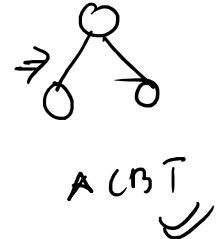
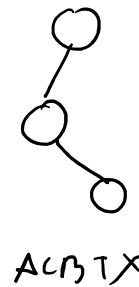
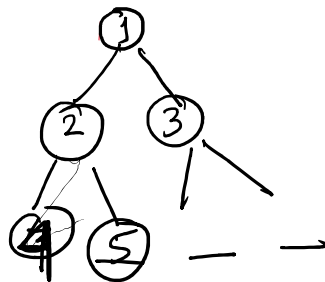
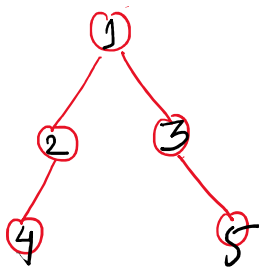
Binary Tree - Tree  $\rightarrow$  nodes  $\rightarrow$  0/1/2 children.

leaf node - Node <sup>with</sup> 0 children



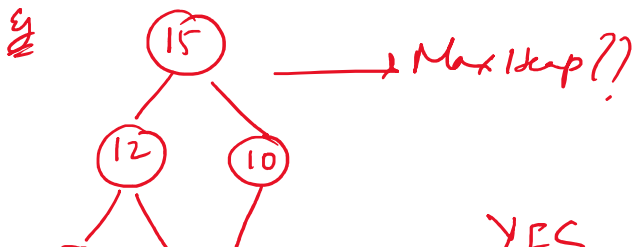
## ~~ACB~~ Almost Complete Binary Tree -

Nodes insert karo  $\rightarrow$  left direction se karo.



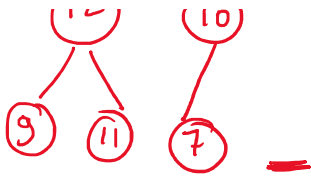
Heap ① ACBT

② Max heap (parents > child)  
Min heap (parents < child)

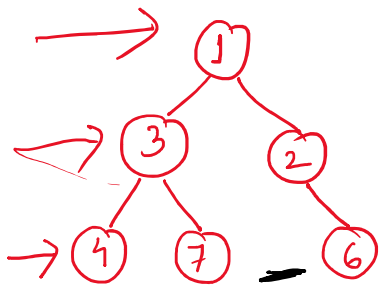


parents.val > child.val.

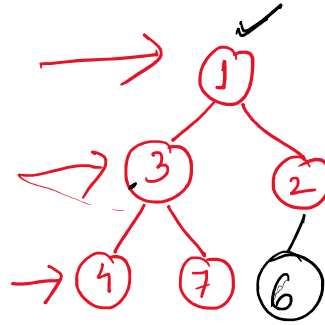
YES



YES



Min Heap ??

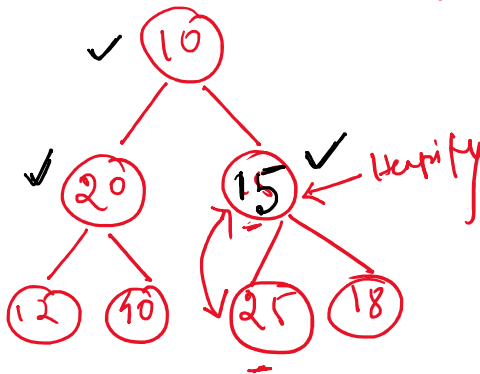


Heapify - Array  $\xrightarrow{\text{convert}}$  Max/Min Heap

Convert to Max Heap:-

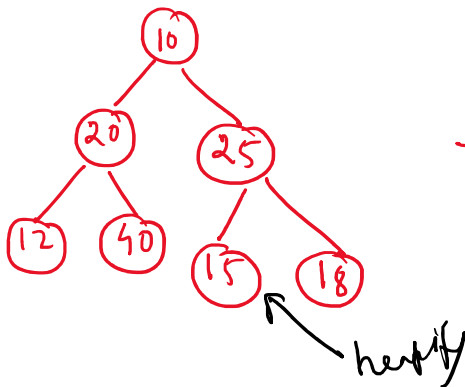
10 20 15 12 40 25 18

→ Represent in the form of ACBT.

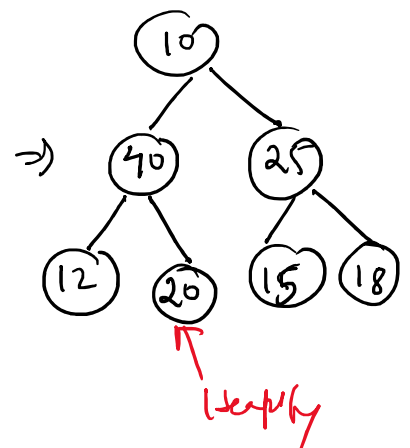
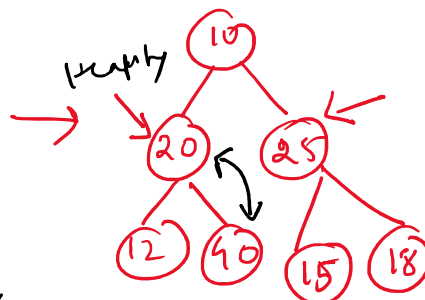


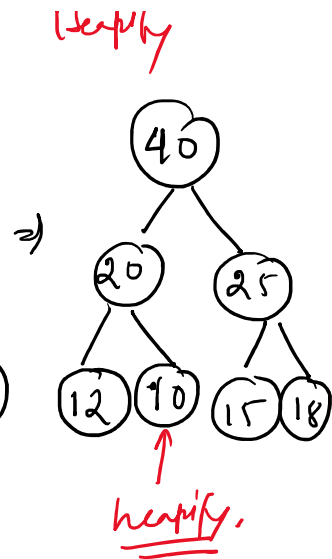
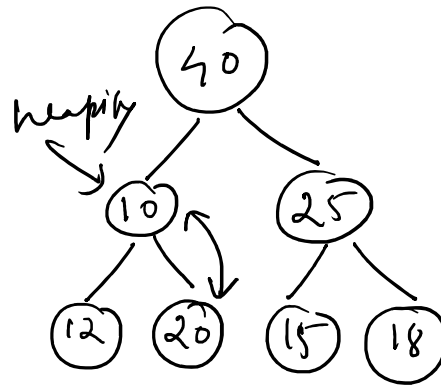
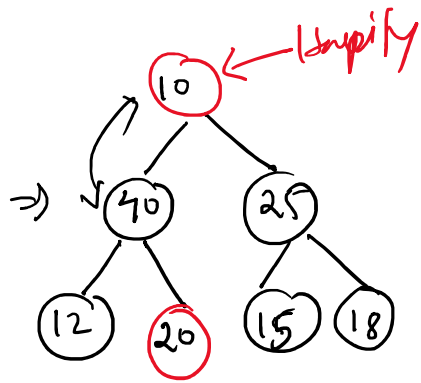
① Start from last non-leaf node.

② parent > child.



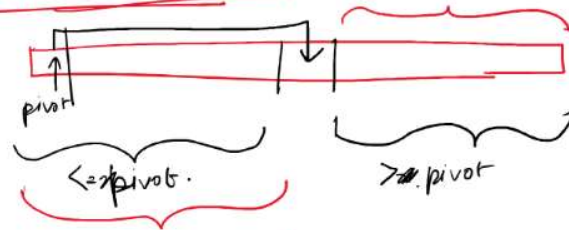
10 ← Heapify





H/W -- Convert array to min heap.

Quick Sort TC & SC.



Best Case / Avg

① Pivot  $\rightarrow$  original position  
 $\rightarrow$  middle

$n/2$

$n/2$

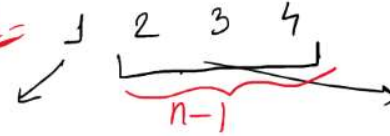
partitioning

$$T(n) = 2T(n/2) + n$$

$$= O(n \log n)$$

Worst Case

eg =



pivot ele = 1

sort

1 2 3 4

$$T(n) = T(n-1) + n$$

$$TC = O(n^2)$$