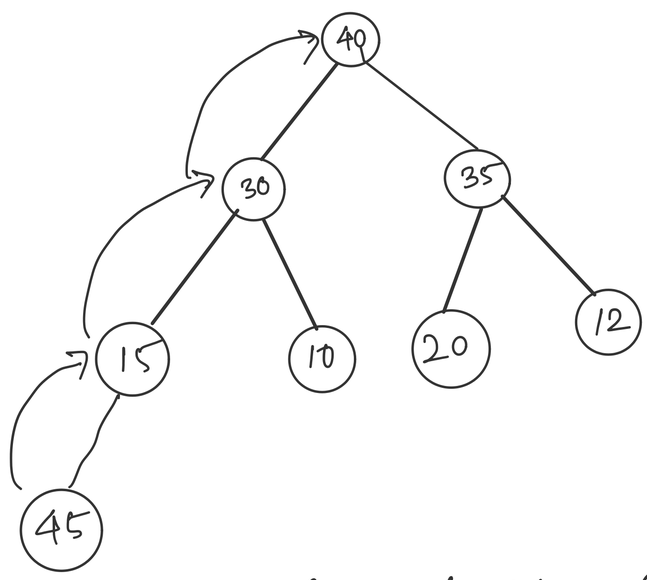
Heapify

05/06/22 2:51 AM

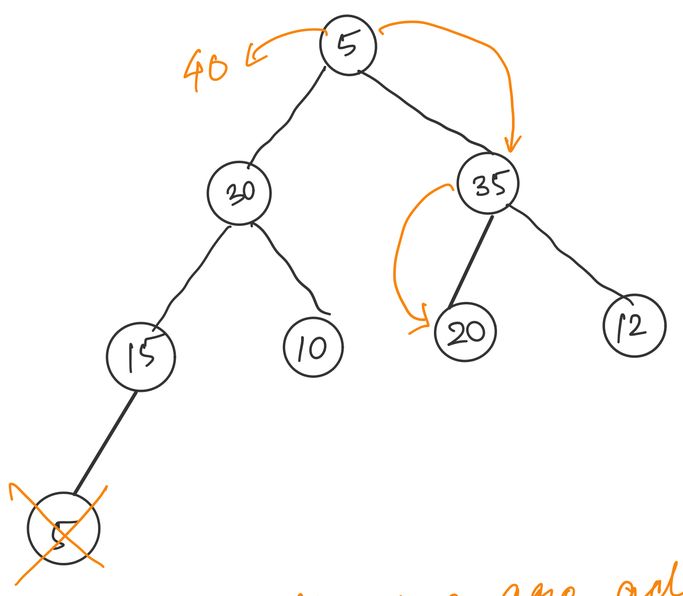
It is a procedure related to creation of heap.

Insert



In insertion elements are adjusted by sending the elements from leaf towards root.

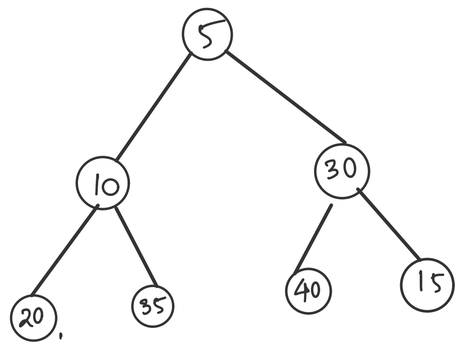
delete

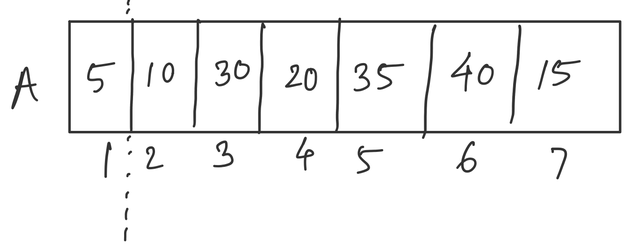


In deletion we are adjusting the elements by bringing the elements from root towards leaf.

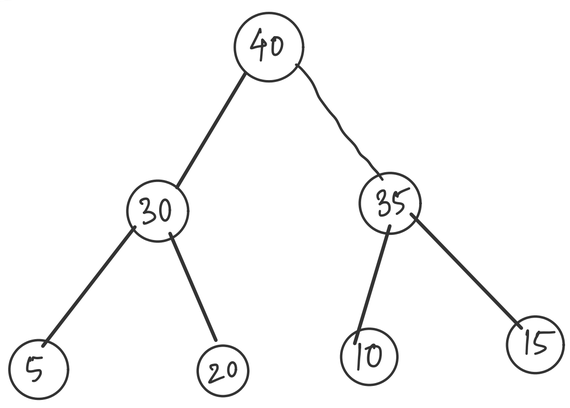
Heapify

creat Heap

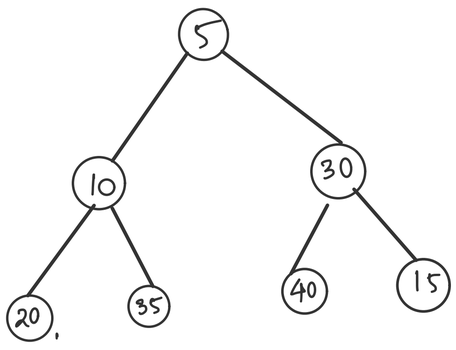


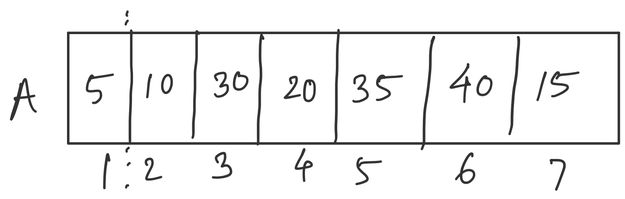


Adjustment ⇒



O(n log n)





we will start from 15 (Right to left)
we will work from 15 downwards.

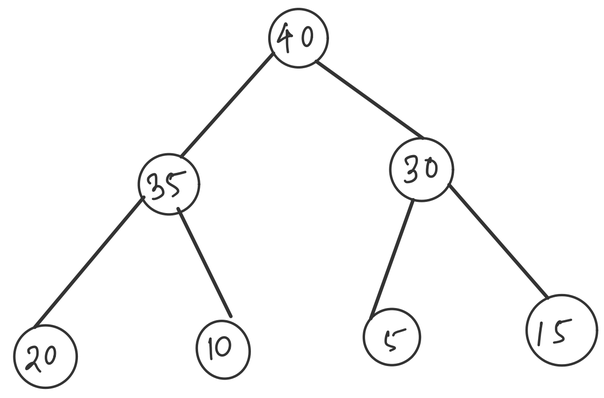
Step 1 ⇒ we will take 15, 40, 35, 20 and look downwards. As there is nothing below them than we dont have to do any modification.

Step 2 ⇒ Come to 30 and compare it with its children. 30 < 40. So interchange 40 and 30

Step 3 ⇒ Same proccedure applied for 10. Intchange 10 and 35.

Step 4 ⇒ Go to 5 and compare with its children. 40 > 5. Interchange 40 and 5.

Step 5 ⇒ Again compare 5 and its children 5 < 30. Interchange 5 and 30.



Max heap

fastest method (least comparison)

This procedure analytically takes O(n) time complexity. because we dont have to process half of the elements i.e the leaf elements.