

HEAP

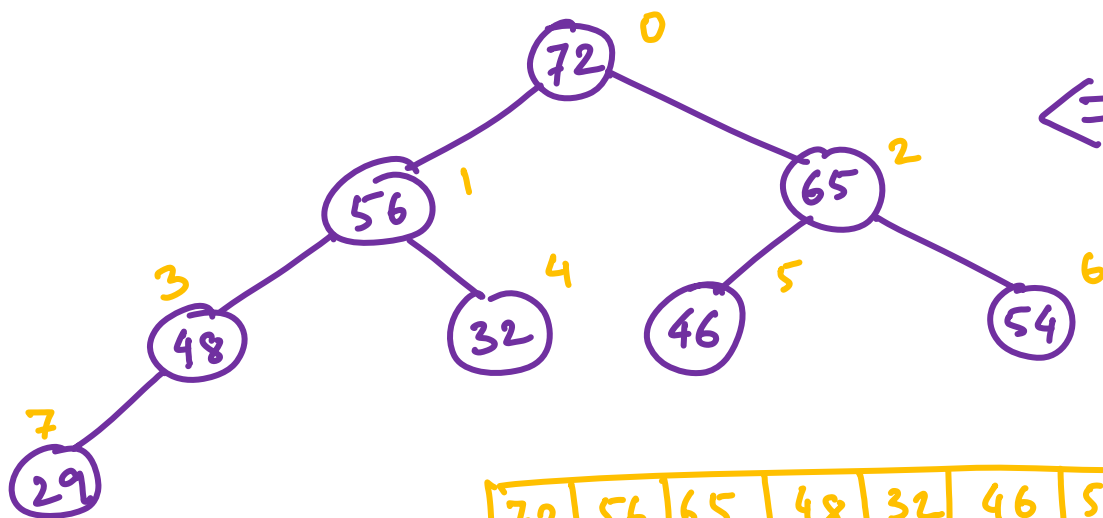
Dated 28/12/2021

It is a complete binary tree.

Complete binary tree {
(a) all leaf nodes are in adjacent levels.
(b) the child positions of non-leaf nodes will be filled up in a left-to-right sequence

Heap is implemented using array rather than linked list.

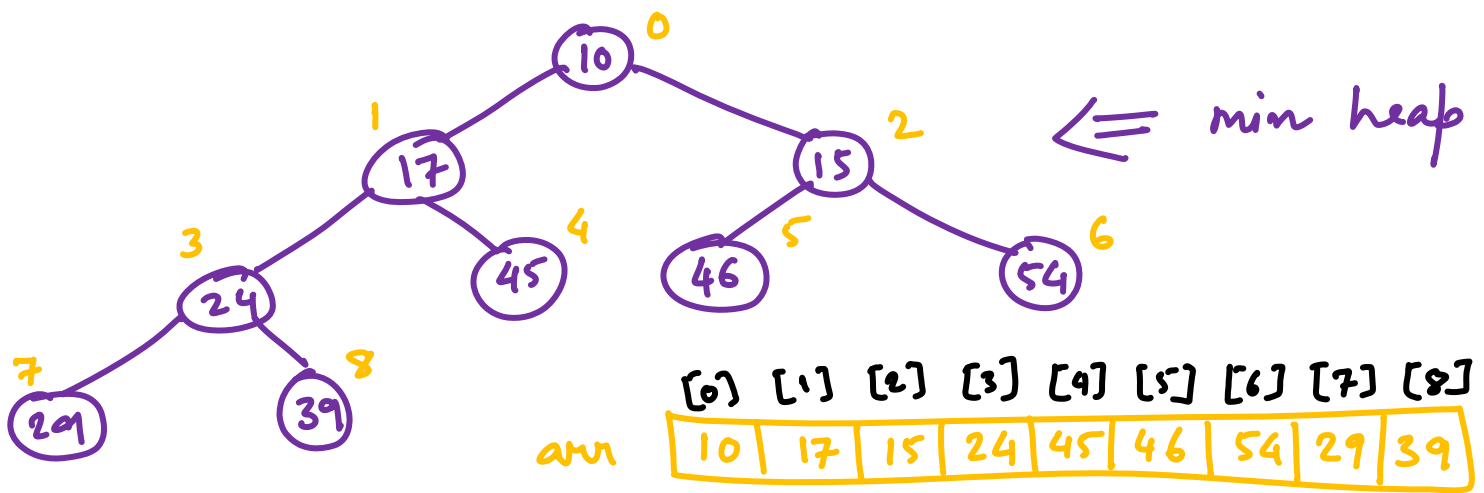
Classification: (1) max heap / descending heap
(2) min heap / ascending heap



⇒ max heap

arr

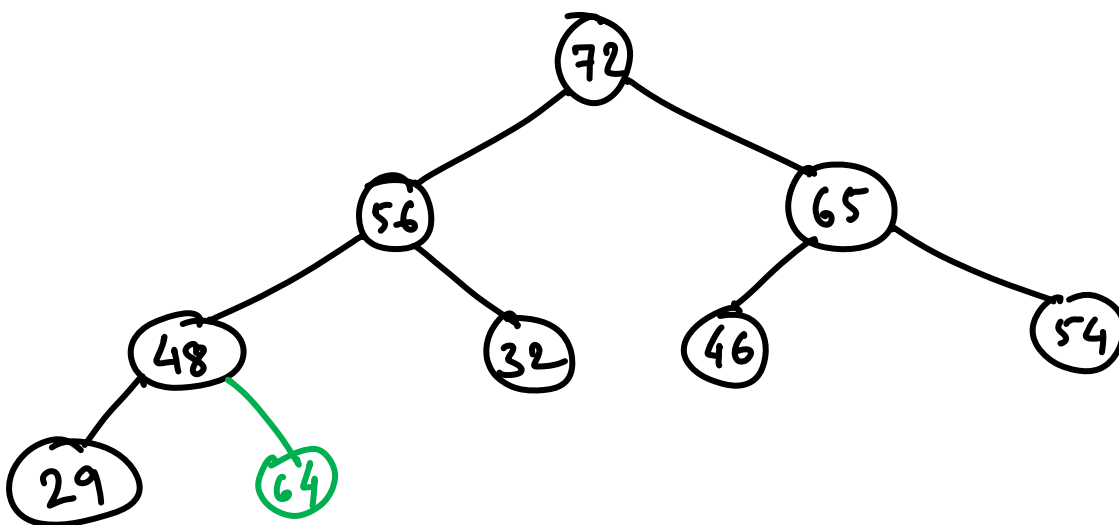
72	56	65	48	32	46	54	29
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]



If $arr[k]$ is the root, then $arr[2k+1]$ and $arr[2k+2]$ are the left and right child respectively.

So, for all further discussion we will be using "max heap".

Insertion with a Heap :-



Let us insert the new item "64" within the above heap.

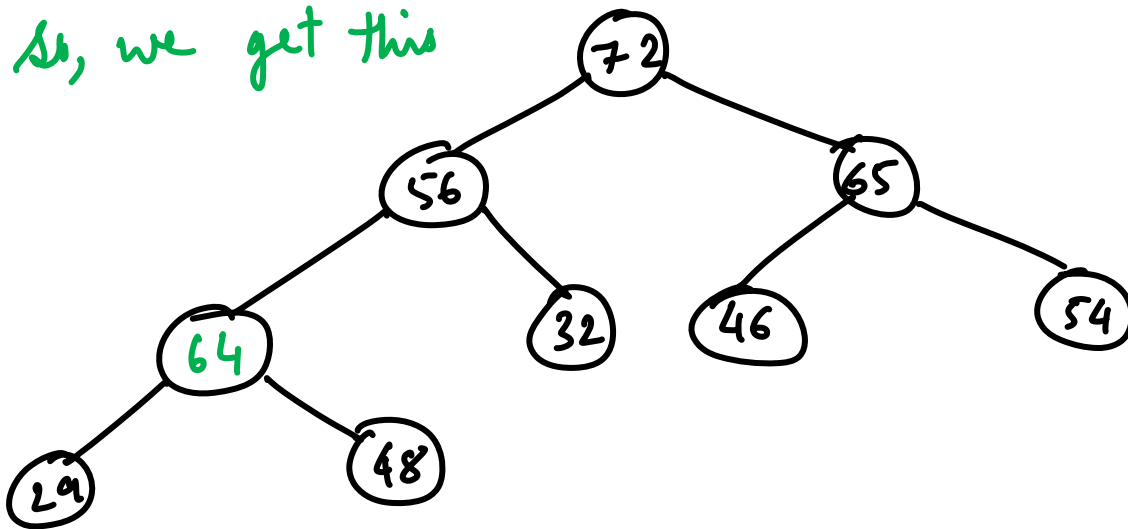
arr

72	56	65	48	32	46	54	29	64
----	----	----	----	----	----	----	----	----

[8]

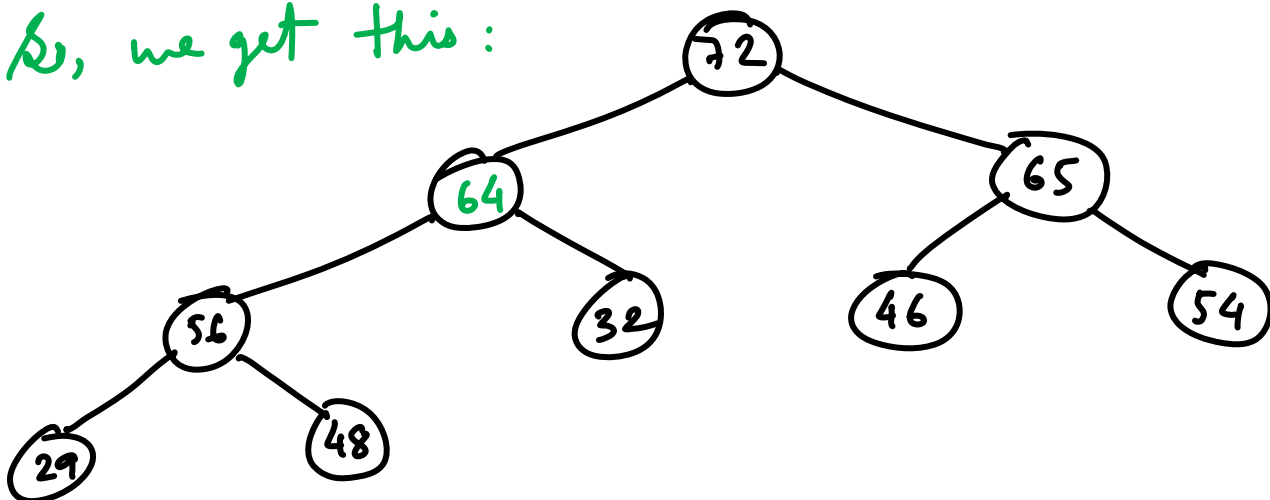
If $64 < 48$,
we find it false.
we need to interchange the position of
64 & 48.

So, we get this



If $64 < 56$, we find this False
therefore, interchange the positions of 64 &
56

So, we get this:



If $64 < 72$, we find this True
So, the proper place/position of "64"
is reached and the insertion is
complete.

arr

72	64	65	56	32	46	54	29	48
[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]

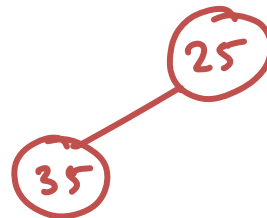
Creation of a Heap :-

The data items in sequence is

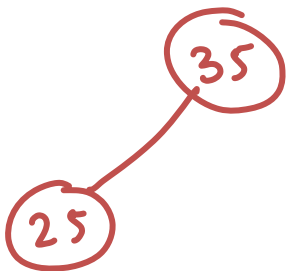
25 35 18 9 46 70 48



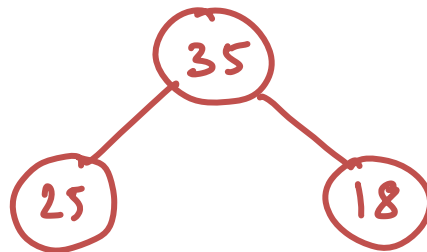
Step - 1



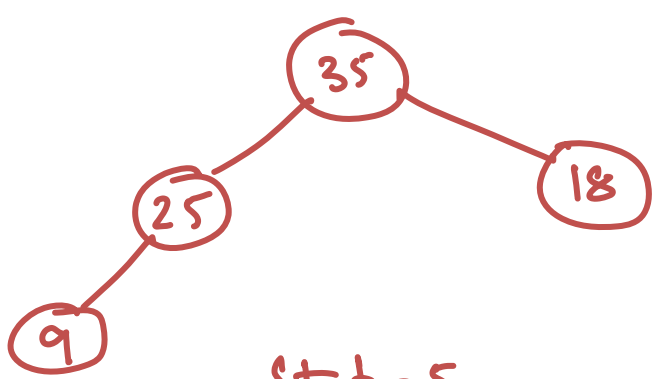
Step - 2



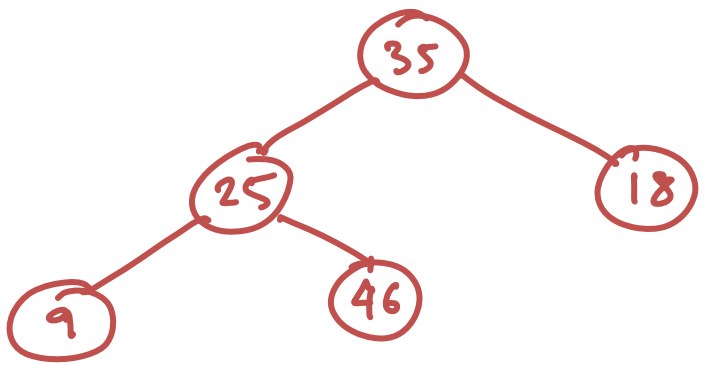
Step - 3 (Interchange)



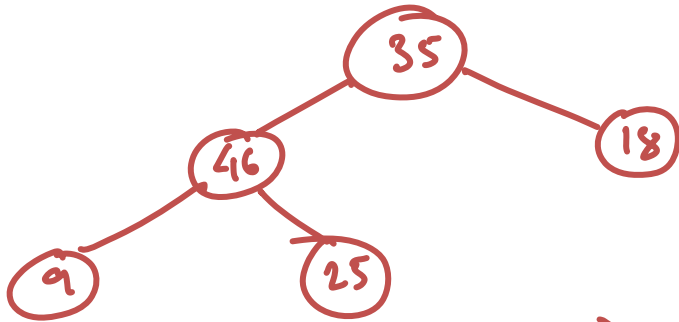
Step - 4



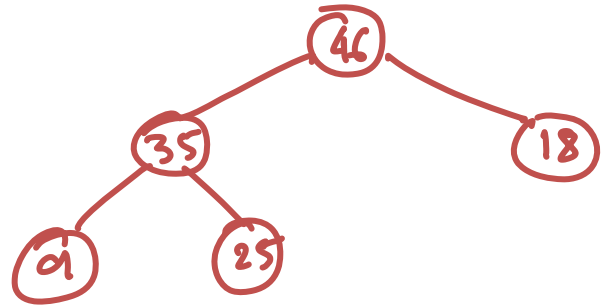
Step - 5



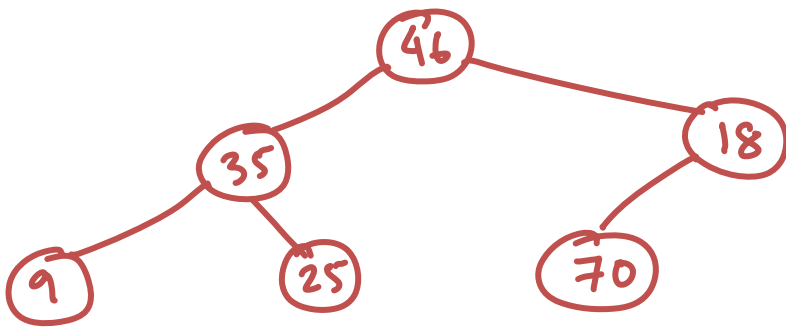
Step - 6



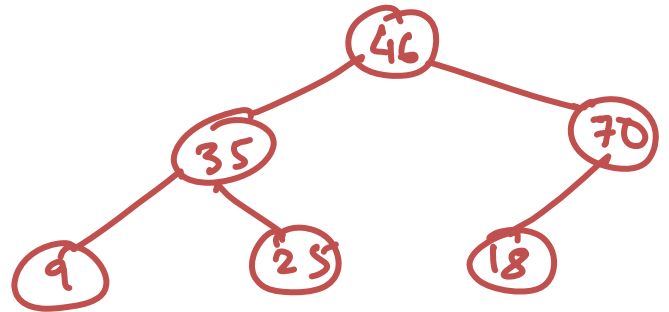
Step - 7 (Interchange)



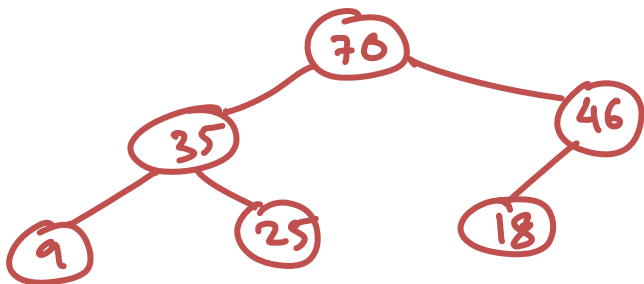
Step - 8 (Interchange)



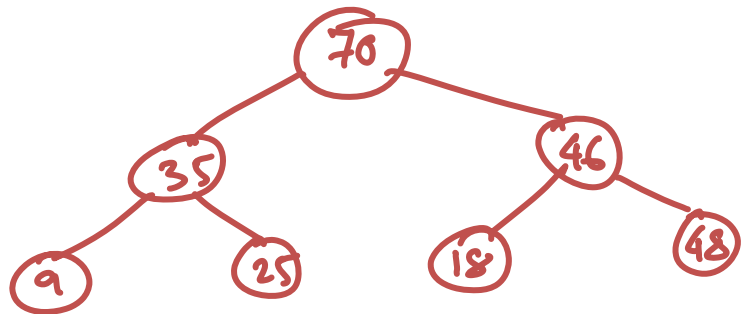
Step - 9



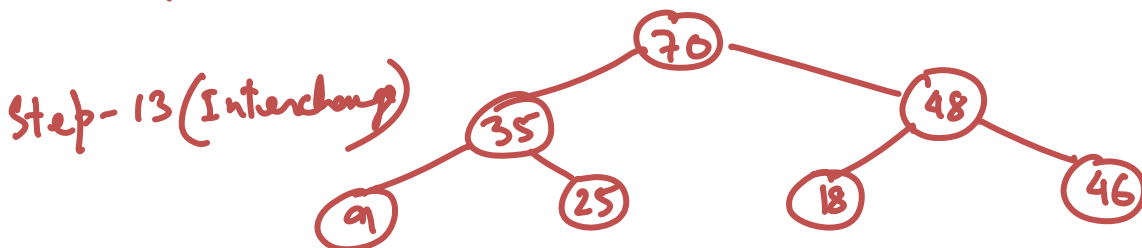
Step - 10 (Interchange)



Step - 11 (Interchange)



Step - 12

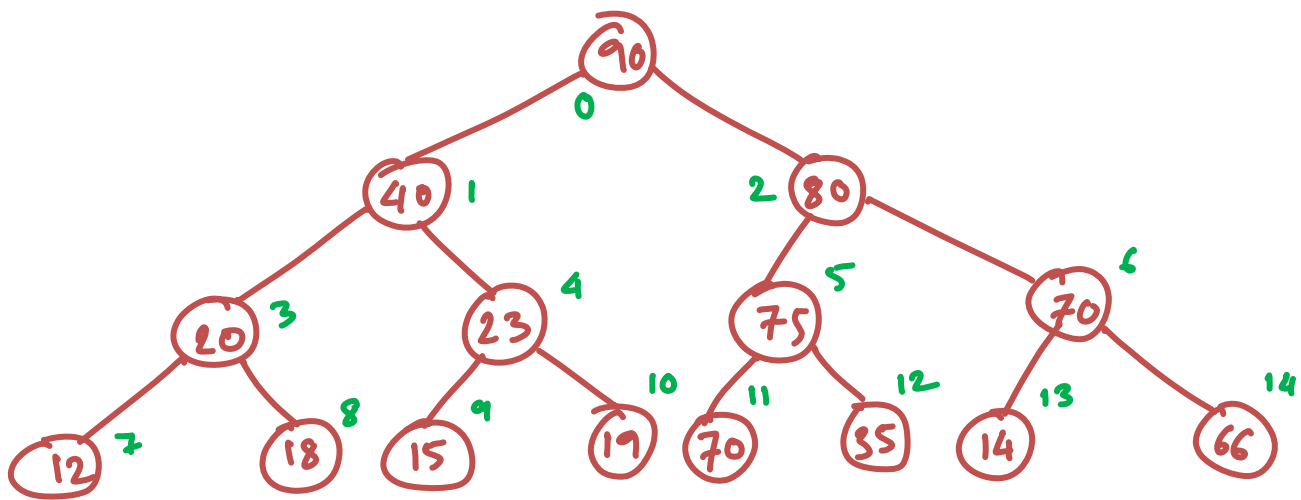


Step - 13 (Interchange)

Deletion within a Heap :- Date 29/12/2021

Deletion operation can be done in a Heap tree as -

- (i) Find the index number of the node to be deleted
- (ii) Replace the deleted node with the last node of the heap.
- (iii) Keep that node at appropriate place.
(node with which the deleted node is replaced)

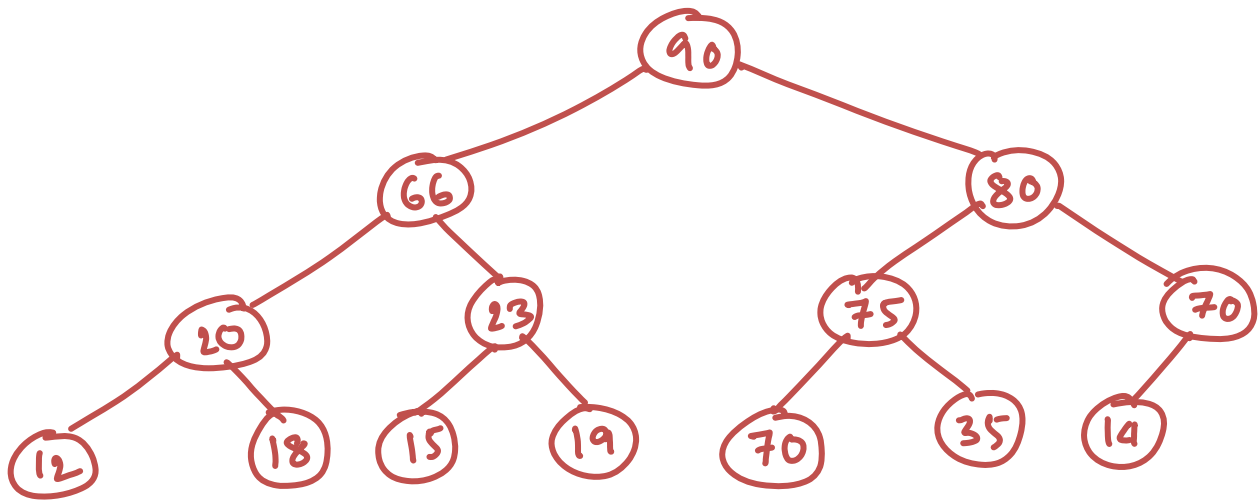


Heap: "H"

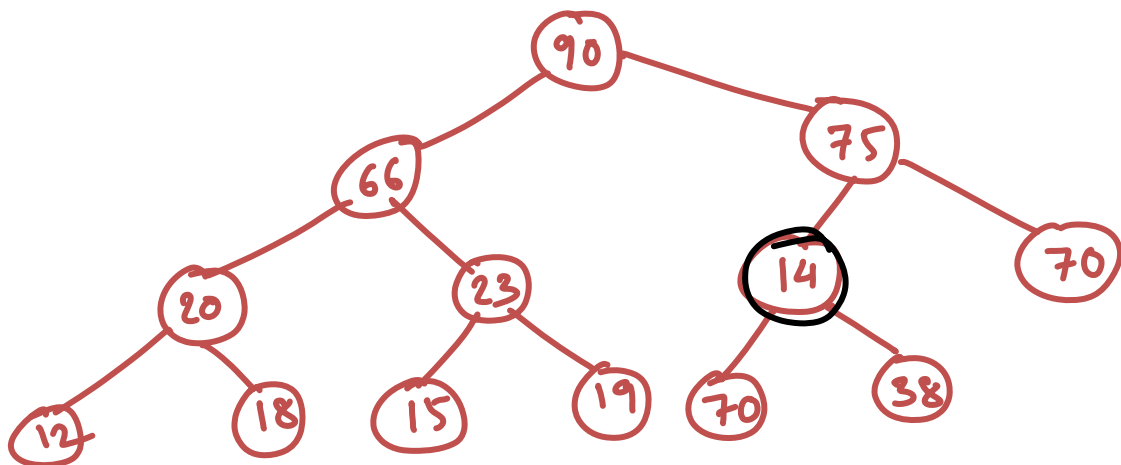
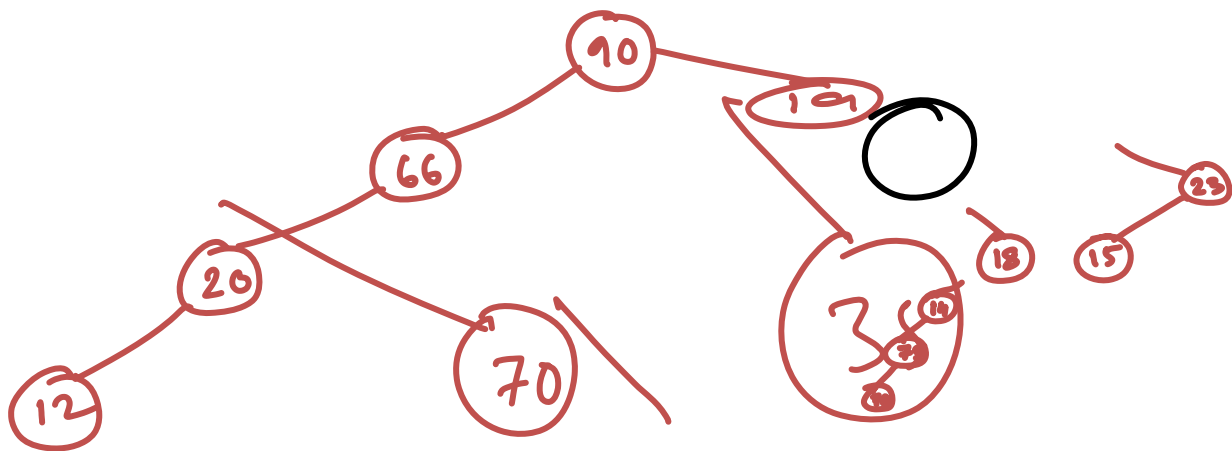
Let us delete node 40 from H

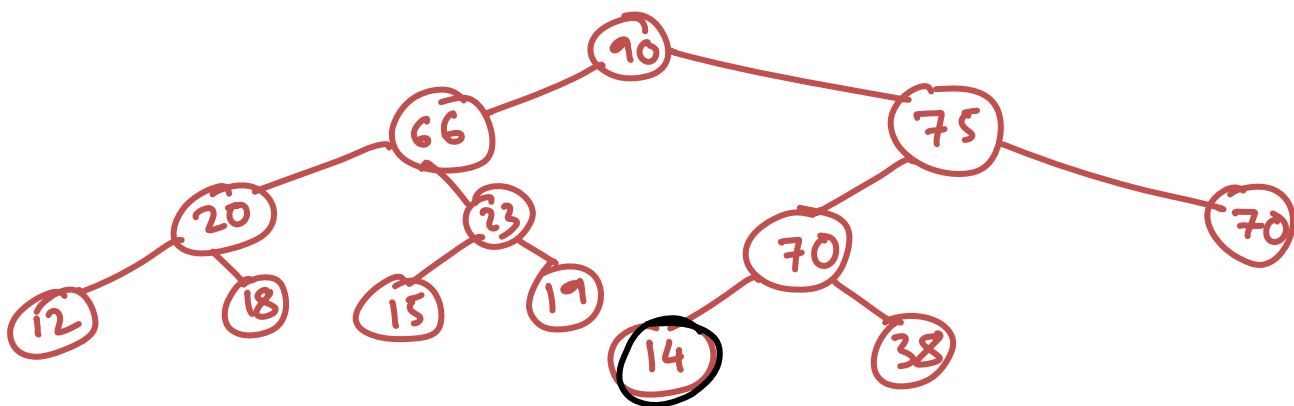
array:

90	40	80	20	23	75	70	12	18	15	19	70	35	14	66
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

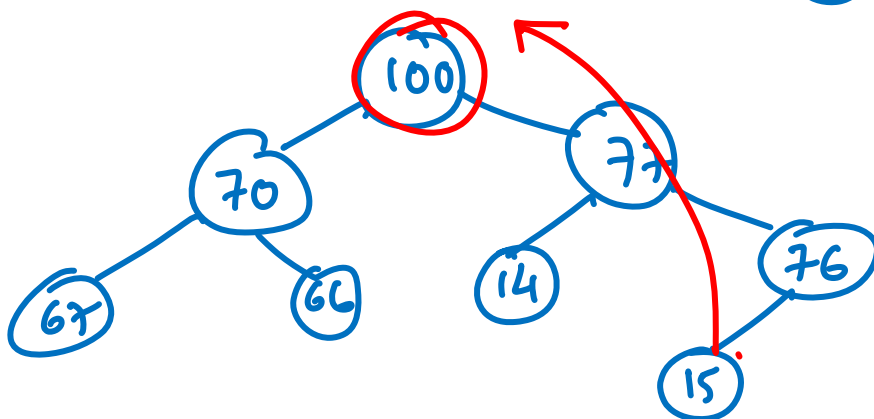
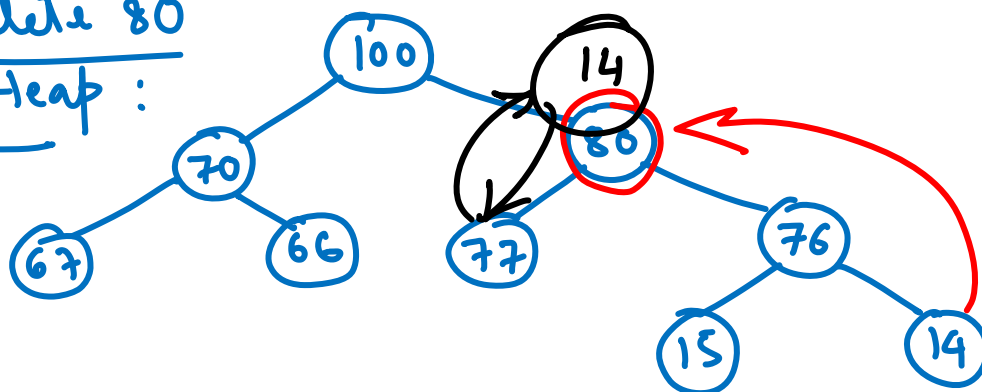


Let us delete "80"





Let us delete 80
from this Heap :



Let us delete 100 from this Heap.

