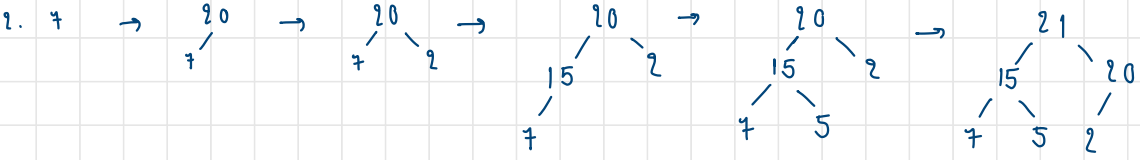
In-lab 10

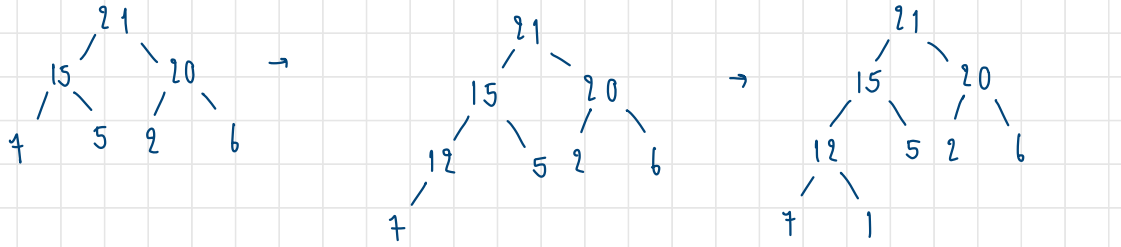
1. for array index starts at index 1

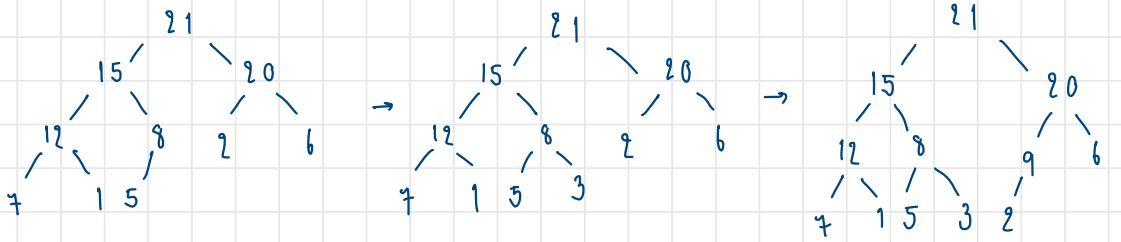
   parent = i/2

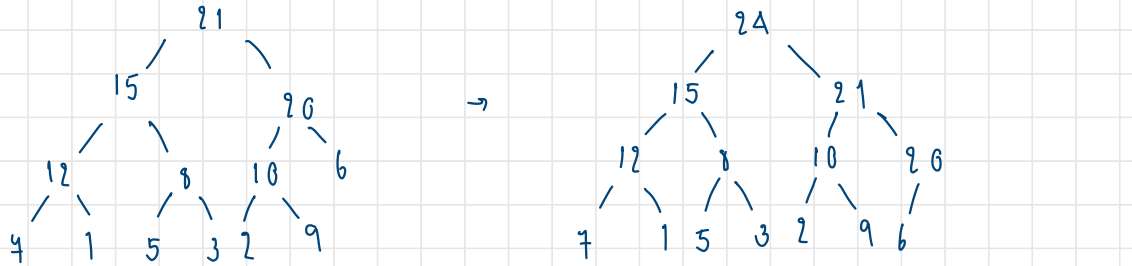
   left child = 2i

   right child = 2i+1

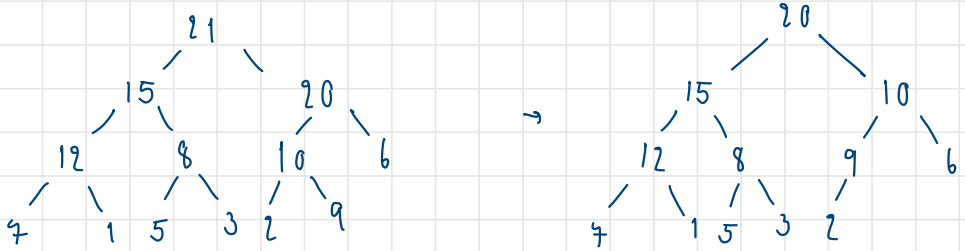
2. 

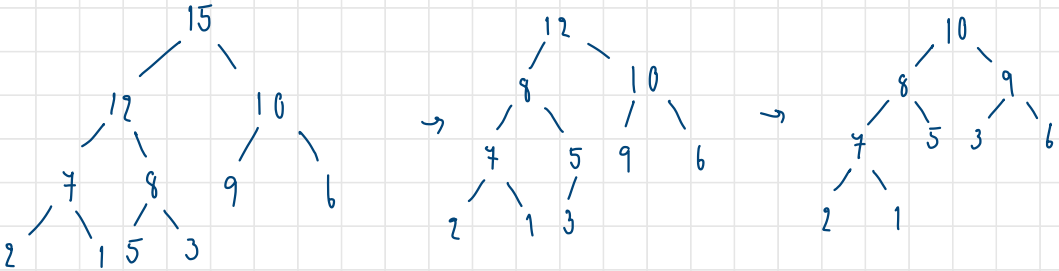


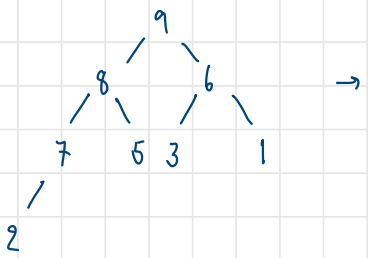




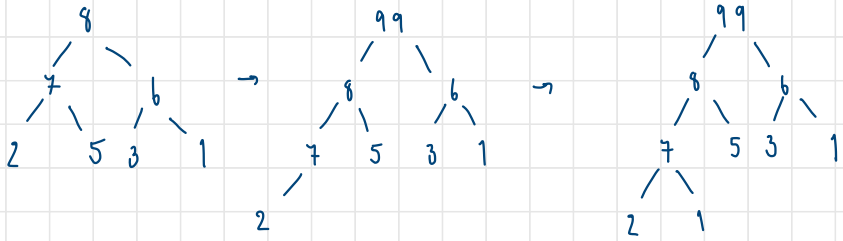
3.

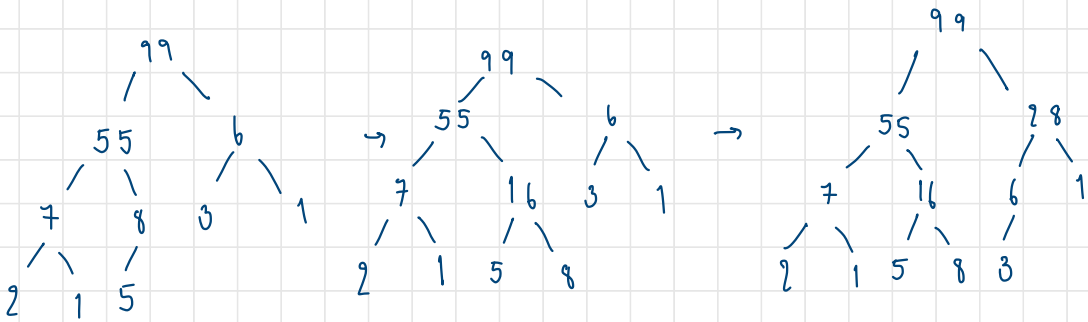

21
15
20
12
8
10
6
7
1
5
3
2
9
20
15
10
12
8
9
6
7
1
5
3
2

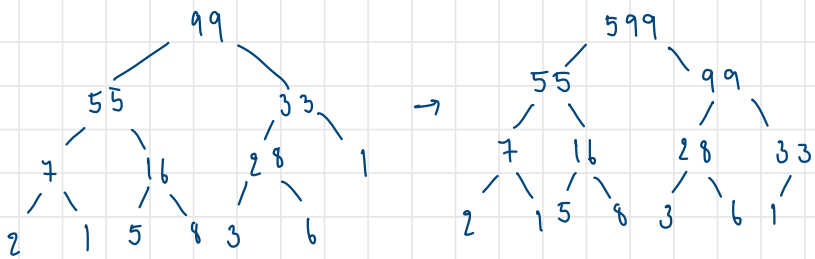

15
12
10
7
8
9
6
2
1
5
3
12
8
10
7
5
9
6
2
1
3
10
8
9
7
5
3
6
2
1

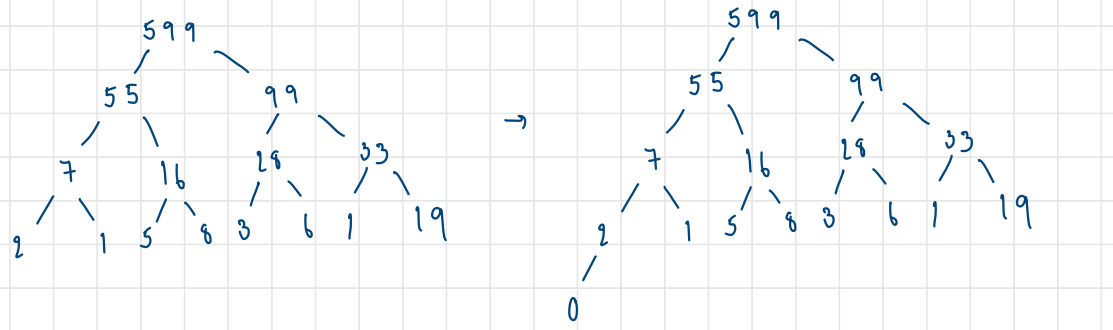

9
8
6
7
5
3
1
2
8
7
6
2
5
3
1

4


8
7 6
2 5 3 1
99
8 6
7 5 3 1
2
99
8 6
7 5 3 1
2 1


99
55 6
7 8 3 1
2 1 5
99
55 6
7 16 3 1
2 1 5 8
99
55 28
7 16 6 1
2 1 5 8 3


99
55 33
7 16 28 1
2 1 5 8 3 6
599
55 99
7 16 28 33
2 1 5 8 3 6 1


599
55 99
7 16 28 33
2 1 5 8 3 6 1 19
599
55 99
7 16 28 33
2 1 5 8 3 6 1 19
0

5. greatest number of nodes = 2^n

6. max. number of node = $2^{n+1} - 1$ no of level = height + 1

7. $O(1)$

8 $O(\log n)$

9 $O(\log n)$ because total time that we have to swap the node would be equal to the height of tree.