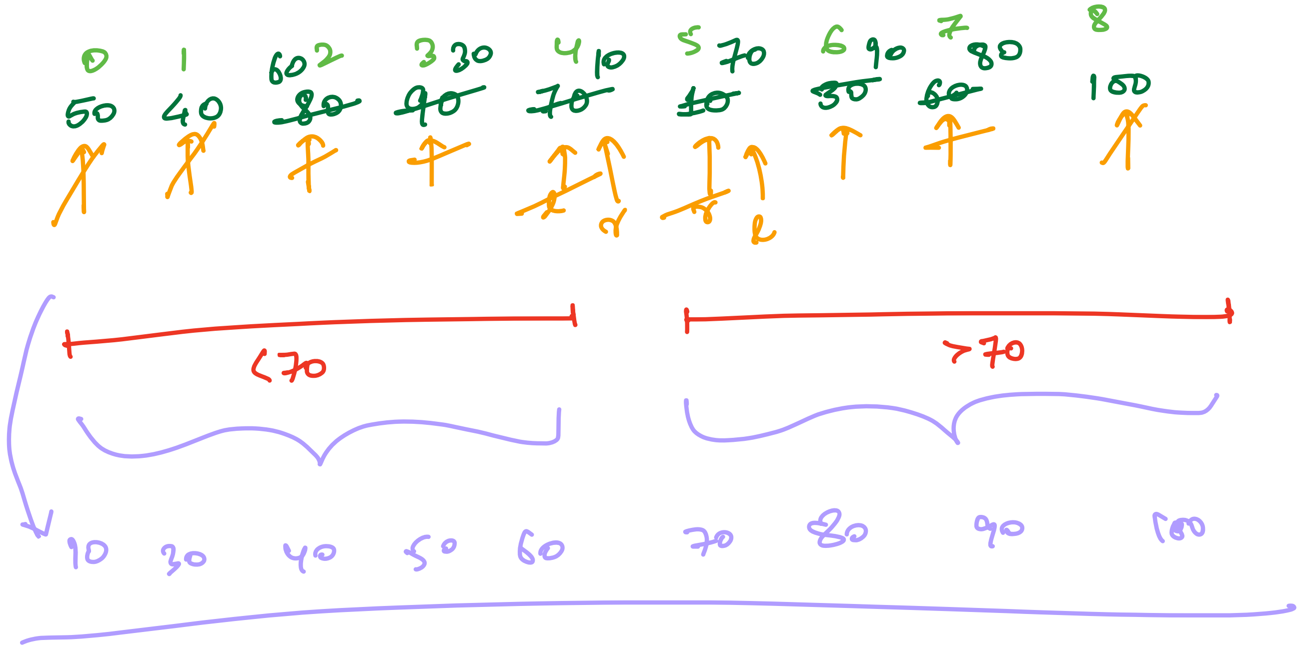
Quick sort

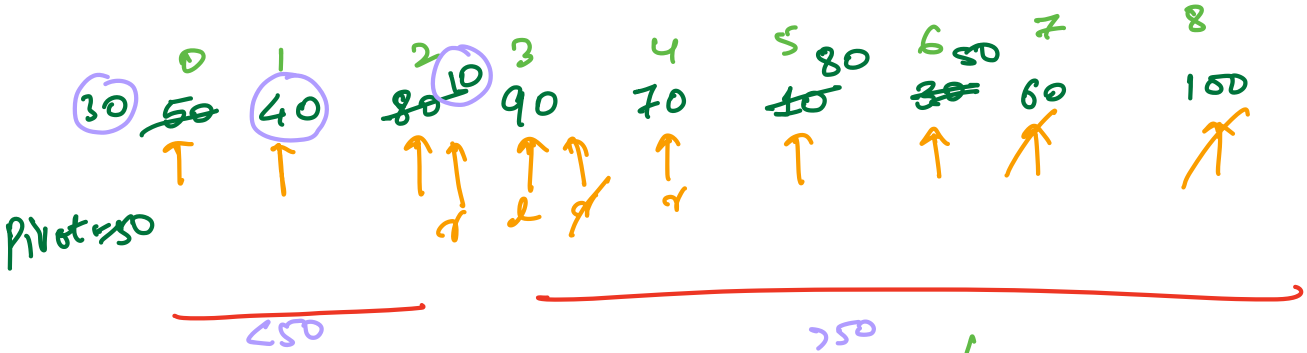
0	1	2	3	4	5	6	7	8
50	40	80	90	70	10	30	60	100

Pivot = 70

Partitioning: <70 >70

$T(n) = T(\frac{n}{2}) + T(\frac{n}{2}) + n$





$T(n) = T(\frac{n}{3}) + T(\frac{2n}{3}) + n$

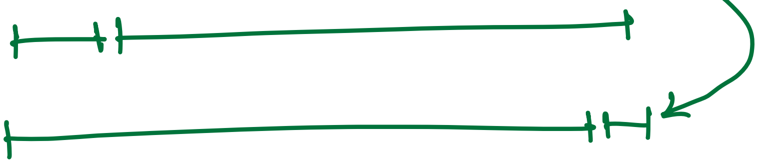
TC: BC: median pivot

n/2 n/2

$T(n) = 2T(\frac{n}{2}) + n$ O(n log n)

WC:

Smallest, largest



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

$$T(n) = T(\cancel{n-1}) + n$$

$$T(\cancel{n-1}) = \cancel{T(n-2)} + (n-1)$$

$$T(n-\cancel{2}) = T(\cancel{n-3}) + (n-2)$$

$$\vdots$$

$$T(\cancel{1}) = 1$$

$$T(n) = n + (n-1) + (n-2) \longrightarrow 1$$

$$= n^2$$

$$1/50 + 1/50 = 2/50 = 1/25$$

Randmizd Quick sort

Partitioning

kth elmt CP ?

kth largest ?

Heap/Priority Queue

Queue:

fifo

PQ:

insert

↳ highest priority

Priority

Marks
higher value higher priority
(Max Heap)

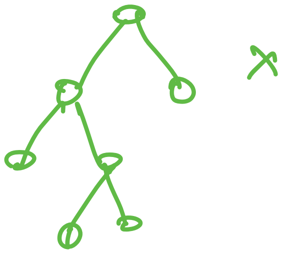
Rank
Lower value higher priority
(Min Heap)

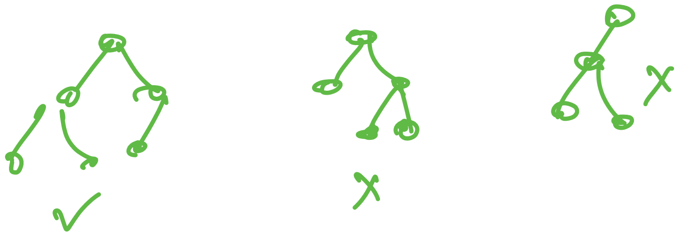
Heap:

- CBT
- Parent Priority > Child Priority

Max: Parent Value > Child Value
Min: Parent Value < Child Value

CBT: n levels
n-1 levels completely filled
nth level left → right





Add: O(log n)
Remove: O(log n)

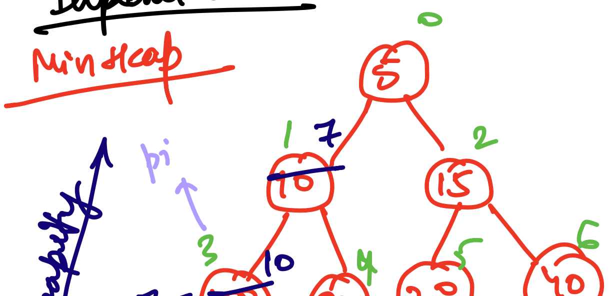
[Add logn
[Remove logn

logn

CBT

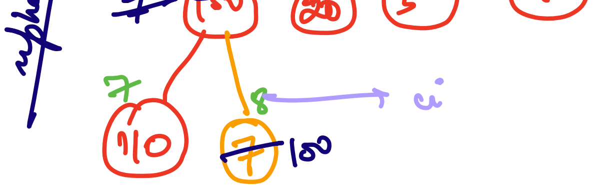
Implementation:

Min Heap



0	1	2	3	4	5	6	7	8	
5	10	15	100	20	30	40	110	7	

note lc: 5



2 →

→ r:6

3 → 7

→ 8

$lci = 2pi + 1$

$rci = 2pi + 2$

$pi = \frac{ci - 1}{2}$