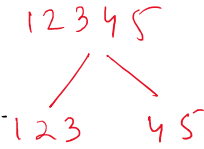
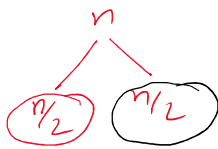


Merge Sort-

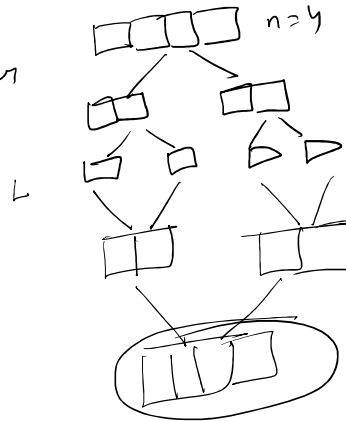
$$T(n) = T(n/2) + T(n/2) + n$$

$$\boxed{T(n) = 2T(n/2) + n} \rightarrow O(n \log n)$$

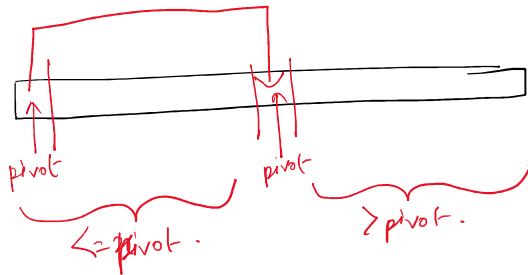
SC $\rightarrow O(n)$



Quick Sort- Based on divide & conquer technique



5 4 3 2 1
↓
4 3 2 1



pivot element.
↓
3 4 2 1 choose
Karrn.
↓
1st / Last element.

3

35 50 15 25 80 20 90 45

pivot = 35

P

9

$p \rightarrow arr[p] \leq pivot$
 $q \leftarrow arr[q] > pivot$

35 20 15 25 80 50 90 45

① Chk if p & q have crossed each other

Swap(pivot, arr[q]);

35 20 15 25 80 50 90 45

② Chk if p & q have not crossed each other =

Swap(arr[p], arr[q])

25 20 15 35 80 50 90 45

≤ 35

OP

> 35

left

right

left: 25 20 15

pivot = 25

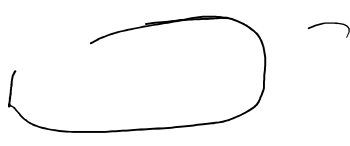
Left 25 20 15

pivot = 25

$arr(p) \leq pivot$
 $arr(q) > pivot$

swap(pivot, arr(q))

25 20 25
 ≤ 25 op



Right

80 50 90 45
~~P~~ ~~P~~ P P

pivot = 80

80 50 45 90
~~P~~ ~~P~~ P P

45 50 80 90
 ≤ 80 op > 80

SC - $O(1)$

TC - Best Case

pivot element \rightarrow Original Position \rightarrow
middle pc.

Worst Case

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

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

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