Recursion - Quick Sort

5, 4, 3, 2, 1

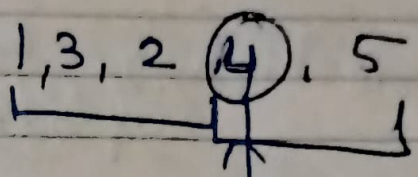
pivot :

Choose any element

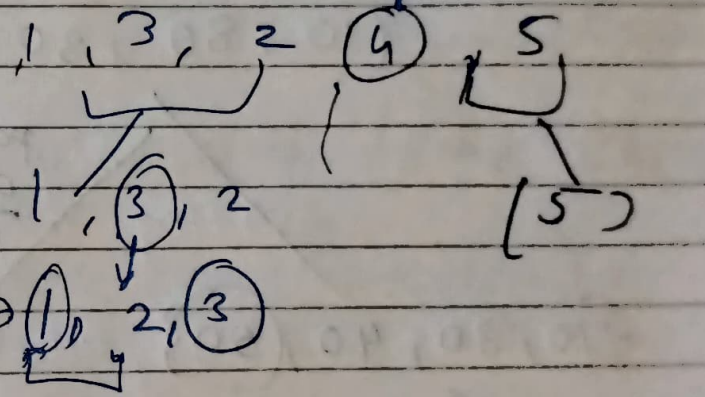
→ after first pass all elements < P will be on LHS of P

&

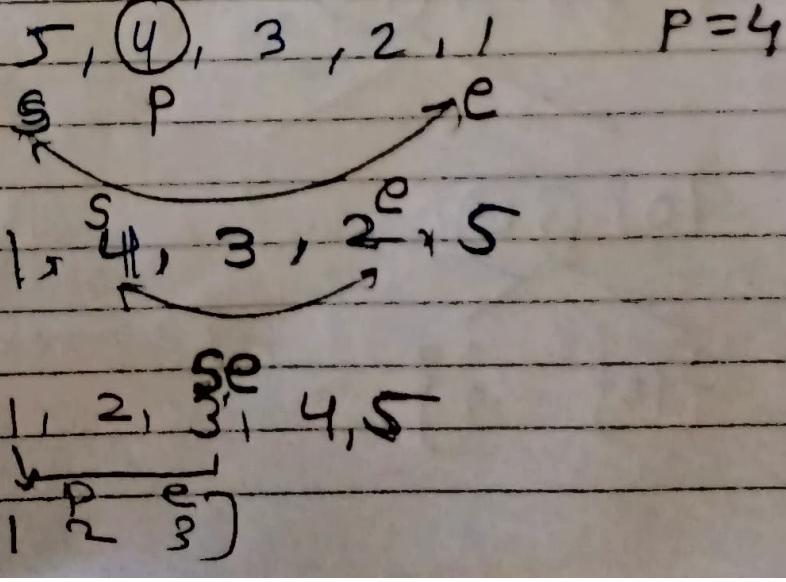
→ > P will be at RHS of P

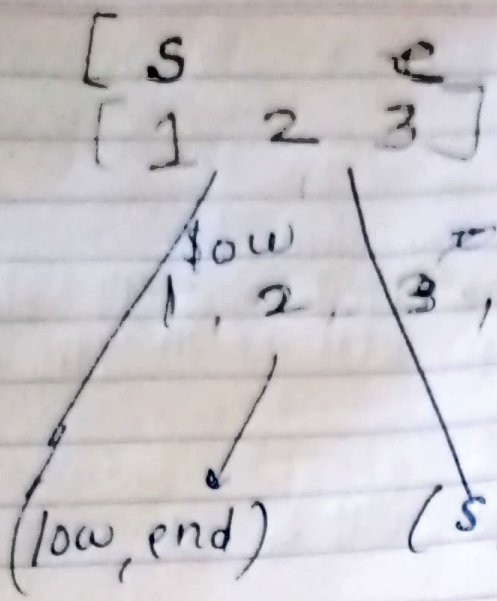


pivot correct position

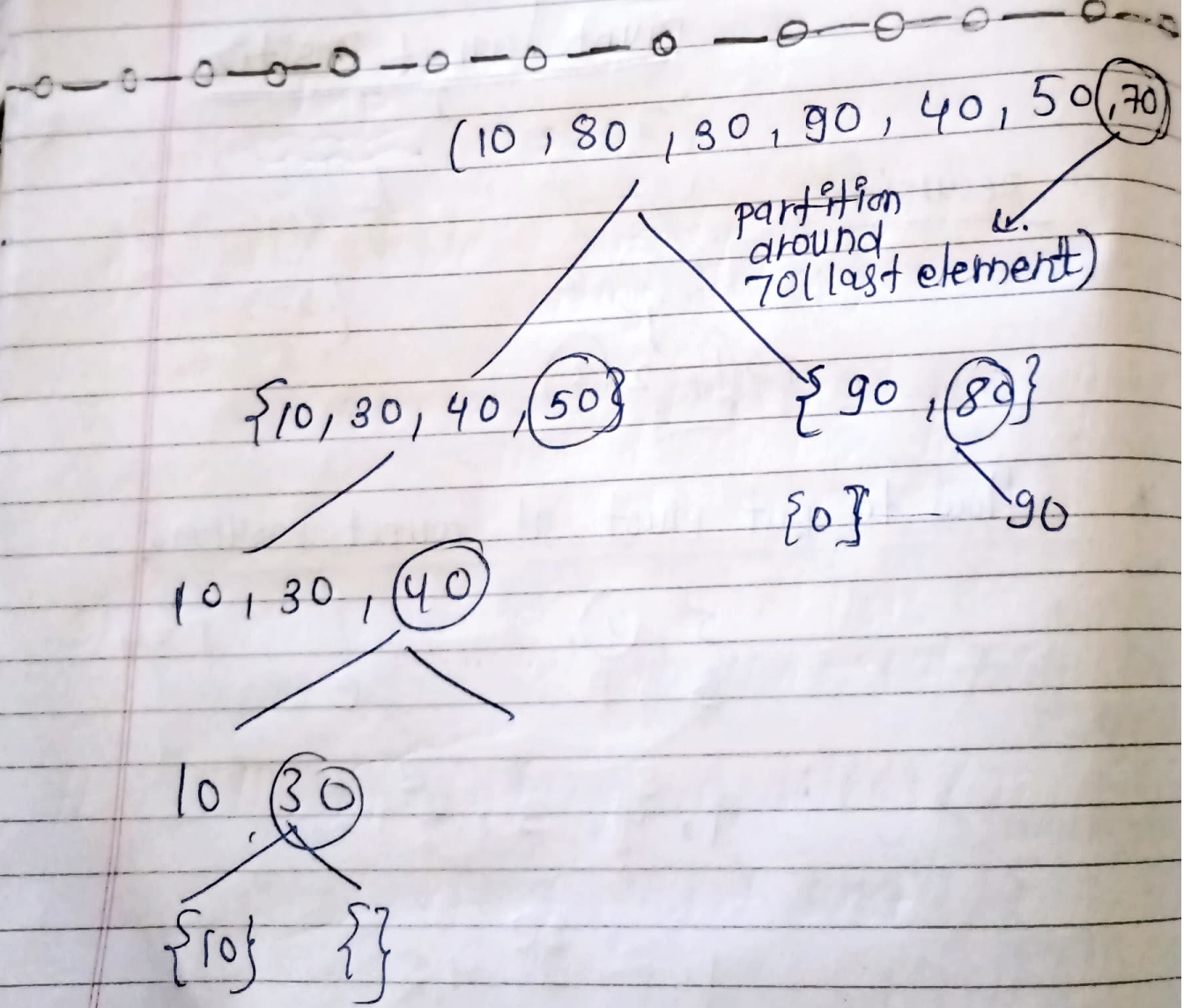


How to put pivot at correct position ?



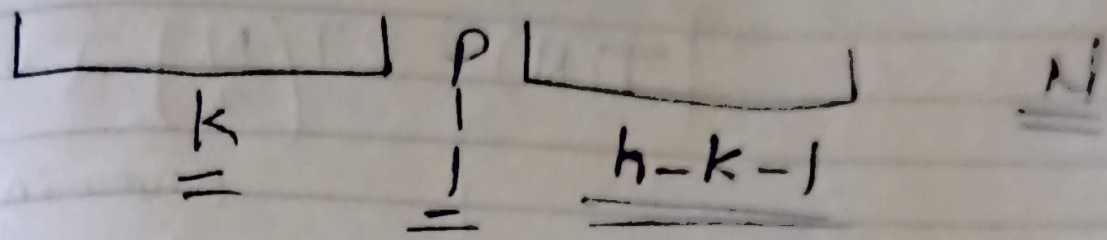


low, high → tells us which part of array you working on
— swap.



How to pick pivot:

* Random element
* corner element
* pick the middle element

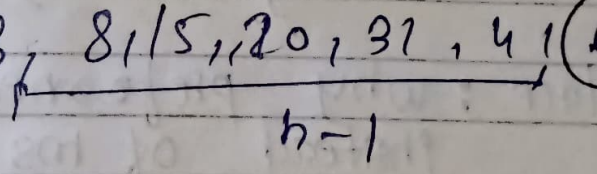


Recurrence relation of quick sort

$$T(N) = T(K) + T(N-K-1) + O(N)$$

Worst case (common sense) General.

k=0



$$T(N) = T(0) + T(N-1) + O(N)$$

$$T(N) = T(N-1) + O(N)$$

worse case complexity : $O(N^2)$ => complexity less

* Best case :

$k = \frac{N}{2}$

$T(N) = T\left(\frac{N}{2}\right) + T\left(\frac{N}{2}\right) + O(N)$

$$T(N) = 2T\left(\frac{N}{2}\right) + O(N)$$

(~~covered in~~)

Best case = $O(N \log N)$

IMP

1) Not stable : (already concered in intial sorting videos)

2) In place : why preferred for arrays instead of ms. ms takes $O(N)$ extra space.

3) MS is better in linked lists done to momory allocation → not continous

Hybrid Sorting algorithms (Tim Sort)

merge sort + insert sort