Quick sort & Bubble sort

Quick sort is an algorithm of the divide and loger type. The problem ~~b~~ will be sorted into two smaller sets

Logic

30	20	10	5	9	17

pivot = 30

	30	

value < 30 value > pivot

for General case

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

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

Explanation with Example

A →	0	1	2	3	4	5	6
	35	50	15	25	80	20	90

A[i] ↗ A[j] ↗

If A[i] > pivot then increament i

here 50 > 35 (stop here)

Check $A[j] < pivot$ decreament j.

$20 < 35$

Now swap the data

$20 \leftrightarrow 50$

Now check again $A[i] > 35$ & $A[j] < 35$
i++ j--

here If i & j cross then swap pivot with A[j].

Analysis of Quick Sort

Time complexity

Worst Case - It occurs when the starting or pivot is largest or smallest element

$T.C = O(n^2)$

Best case - It occurs when the pivot is near of middle or middle element.

$T.C \rightarrow \log n$

Average case - When above two condition not occur

$T.C = (n \log n)$

Bubble Sorting

Logic

compare first two elements and if the left element is greater than right element, then swap their position and continue the same till the end of Array.

Algorithm

A is array of values N is the no of elements

```
(1) Repeat for round 1 2 3 --- N-1
(2) Repeat for i=0,1, ---(N-1-round)
(3)   If A[i] > A[i+1] then swap
(4) Return.
```

Time complexity = $O(n^2)$

To reduce Time complexity

We have to skip the round that we already sorted.

In this way we get

T.C → $O(n)$ best case.