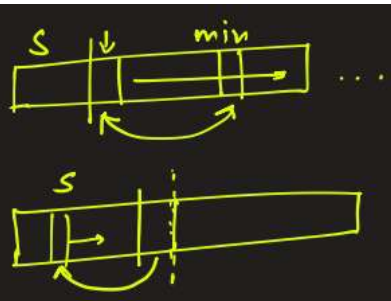
Arrays Lecture 5

Thursday, 18 July 2024 6:06 AM

Insertion Sort



works iteratively inserting each element of an unsorted list into its correct position in a sorted portion of the list.

```
Insertion_Sort (A, n) {
    for (j: 2 → n) {
        key ← A[j]
        for (i: j-1 → 1 && A[i] > key) {
            A[i+1] ← A[i]
            i = i-1
        }
        A[i+1] ← key
    }
}
```

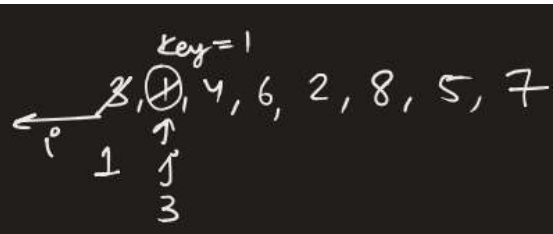
Time :-

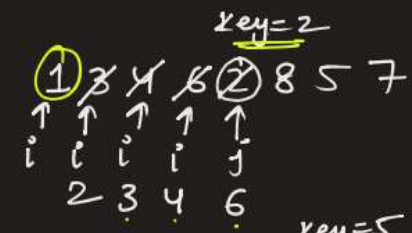
Worst :-	$O(n^2)$
Best :-	$O(n)$
Average :-	$O(n^2)$

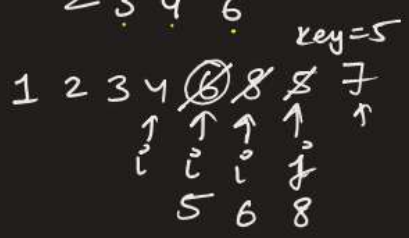
In-place / Out-of-place :- In place

Extra $S = O(1)$

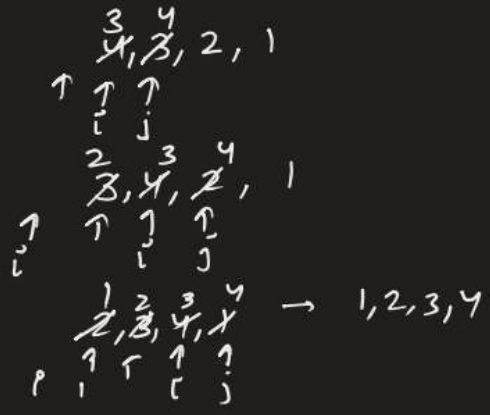
Stable / Unstable :- Stable







↳ 1, 2, 3, 4, 5, 6, 7, 8



=x=

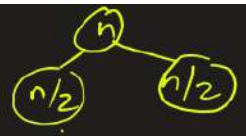
Bubble, Selection, Insertion

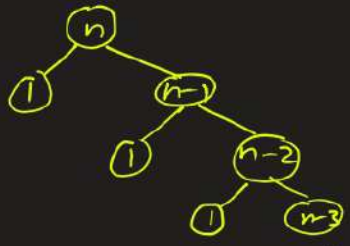
↳ one part of the array sorted

✓ $T(n) = T(n-1) + O(n)$

[Subtract & Conquer] ↗

{ merge, quick





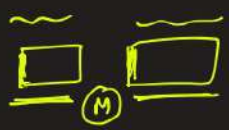
=x=

Merge Sort

↳ Worst :- $O(n \log n)$
Best :- $O(n \log n)$
Avg :- $O(n \log n)$

$O(n)$ –

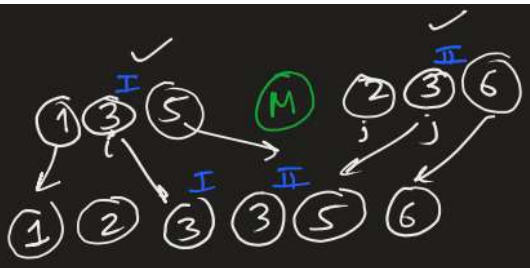
$T(n) = 2T(n/2) + O(n) \rightarrow O(n \log n)$



Inplace / out-of-place :- Out-of-place

Extra space :- $O(n)$

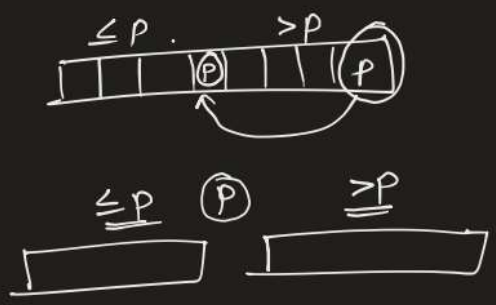
Stable / Unstable :- Stable



we can guarantee stability while merging

$\Rightarrow\times\Leftarrow$

Quick Sort



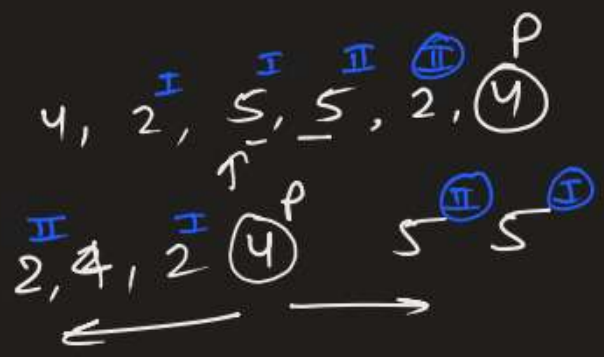
Time:-

Best	$O(n \log n)$
Average	$O(n \log n)$
Worst	$O(n^2)$

Inplace / outplace :- Inplace

Extra Space :- $O(1)$

Stable / Unstable :- Unstable



Heap Sort

↳ Best, Worst, Average → $O(n \log n)$

Inplace Algorithm

Unstable

=×=

The worst-case running times of Insertion sort, Merge sort, and Quicksort, respectively, are:
[GATE CSE 2016 Set 1]

$O(n^2)$ $O(n \log n)$ $O(n^2)$

1. $\Theta(n \log n)$, $\Theta(n \log n)$, and $\Theta(n^2)$.
2. $\Theta(n^2)$, $\Theta(n^2)$, and $\Theta(n \log n)$.
3. $\Theta(n^2)$, $\Theta(n \log n)$, and $\Theta(n \log n)$.
4. $\Theta(n^2)$, $\Theta(n \log n)$, and $\Theta(n^2)$.

(4)

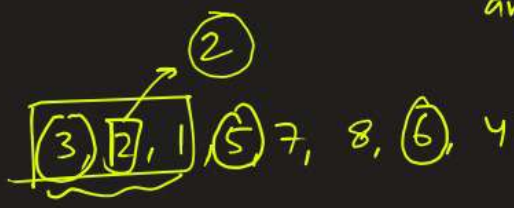
An unordered list contains n distinct elements. The number of comparisons to find an element in this list that is neither maximum nor minimum is

[GATE CSE 2015 Set 2]

1. Θ(nlogn)
2. Θ(n)
3. Θ(logn)
4. Θ(1) ✓

(2) → max & min O(n)

any 3 elements



3, 2, 1 _

3, 5, 6 → cannot be max/min

What is the number of swaps required to sort n elements using selection sort in the worst case?

[GATE CSE 2009]

1. Θ(n) ✓
2. Θ(n log n)
3. Θ(n^2)
4. Θ(n^2 log n)

Complexity :- $O(n^2)$

Comparisions :- $O(n^2)$

Swaps :- O(n)

Which of the following sorting algorithms has the lowest worst-case complexity?
[GATE CSE 2007]

1. Merge sort → $n \log n$
2. Bubble Sort → n^2
3. Quick Sort → n^2
4. Selection Sort → n^2

Which one of the following in place sorting algorithms needs the minimum number of swaps?
[GATE CSE 2006]

1. Insertion Sort → n^2 swaps
2. Quick Sort → $n \log n$ swaps
3. Heap Sort → $n \log n$ swaps
4. Selection Sort → n swaps

heapify

Bubble Sort → n^2 swaps

Merge Sort → No swapping (Extra space)