Introduction to Sorting - 2

24 June 2022 20:10

1. Insertion Sort
2. Stability
3. Problem Solving

Inplace

{ [a,b] = [b,a] }

$O(n^2)$

```
function bubbleSort(arr){
    let n = arr.length; // 5
    for(let i=0;i<n-1;i++){
        var isswapped = false;
        for(let j = 0;j < n-1-i;j++){
            if(arr[j] > arr[j+1]){
                //swap
                [arr[j],arr[j+1]] = [arr[j+1],arr[j]];
                isswapped=true;
            }
        }
        if(!isswapped){
            break;
        }
        console.log(arr)
    }
    return arr;
}

bubbleSort([5,1,4,2,8])
```

[1,2,3,4,5]

Best - O(n)
Worst - $O(n^2)$
Avg - $O(n^2)$
fn)

Insertion Sort-

7 2 4 1 5 3 [7]

In every iteration, we pick 1st elem of the unsorted region and place it at its right position in the Sorted region.

✓ 7 ✗ | 2 4 1 5 3
Sorted | unsorted

elem = 2

✗ 2 ✓ 7 ✗ | 4 1 5 3 — 2 < 7
Sorted | unsorted

I
2 4 7 | 1 5 3 — key = 4
Sorted | unsorted
4 < 7 ✓
4 < 2 ✗
key = 1

II
2 4 7 | 1 5 3
1 < 7
1 < 4
1 < 2

III
1 2 4 7 | 5 3 — key = 5
Sorted | unsorted
5 < 7 ✓
5 < 4 ✗

1 2 3 4 5 7 — key = 3
3 < 7 ✓ 3 < 2 ✗
3 < 5 ✓
3 < 4 ✓

Q 12 | 11 13 5 6 — dry run
Sorted | unsorted

I
11 | 12 | 13 5 6 — key = 11
Sorted | unsorted
12 > 11 ✓

II
11 12 | 13 | 5 6 — key = 13
13 > 12 ✓

III
5 11 12 | 13 | 6 — key = 5
Sorted | unsorted
13 > 5 ✓
12 > 5 ✓
11 > 5 ✓

IV
5 6 11 12 | 13 — key = 6
13 > 6 ✓
12 > 6 ✓ 11 > 6 ✓
5 6 8

Q 3 4 5 6 | 1 — key = 1
Unsorted

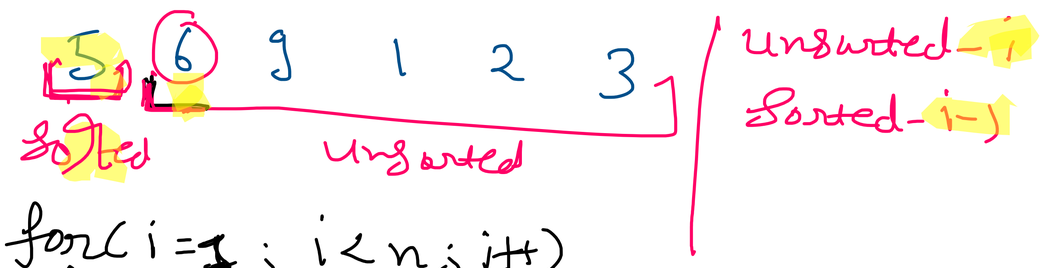
4 Shifts => n-1 shifts

n-1 + n-1 + - - -

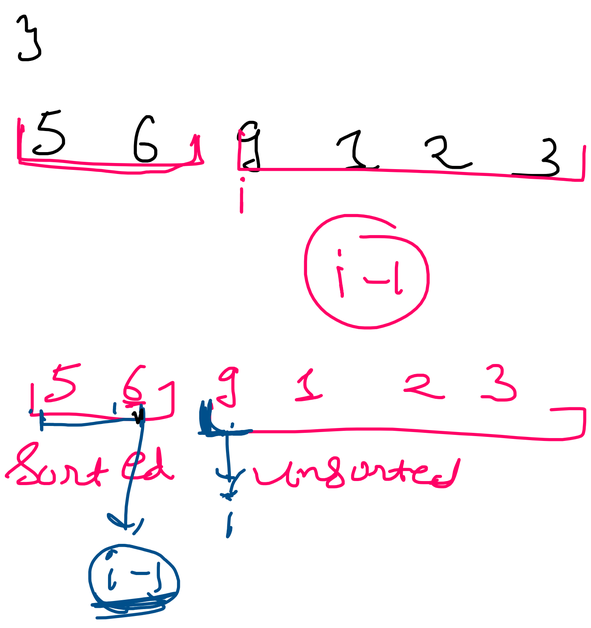
(n-1) → $O(n^2)$

Code

n-1 iterations



for(i=1; i<n; i++)
{



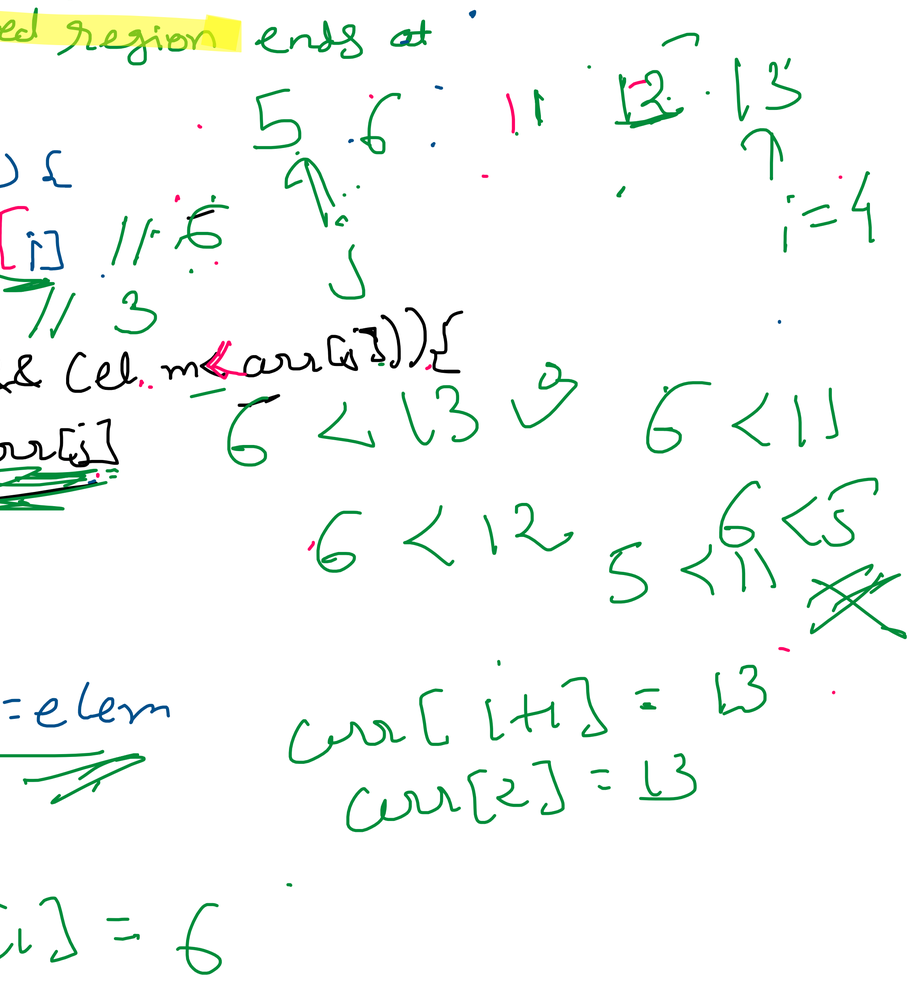
key = 9

9 ≥ 6

9 ≥ 5

If your unsorted region starts at ith index, then sorted region ends at i-1 index.

```
for(i=1; i<n; i++) {
    let elem = arr[i]   // 6
    let j = i-1;        // 3
    while((j>=0) && (elem < arr[j])){
        arr[j+1] = arr[j]
        j--
    }
    arr[j+1] = elem
```



0+1

arr[1] = 6

TC

Best - O(n)

Worst - $O(n^2)$

Avg - $O(n^2)$

Bubble	Selection	Insertion
B - $O(n^2)$ → $O(n)$	$O(n^2)$	$O(n)$
W - $O(n^2)$	$O(n^2)$	$O(n^2)$
A - $O(n^2)$	$O(n^2)$	$O(n^2)$

I.C -> S.C -> O(1)

↳ Inplace Sorting algo

Stability of algo

name	age
John	24
Prakshal	23
Naman	21
Naman	25

Sort by → age

name	age
Naman	21
Prakshal	23
John	24
Naman	25

Sort by → name

John	24
Naman	21
Naman	25
Prakshal	23

John	24
Naman	25
Naman	21
Prakshal	23

Stable Sorting algorithm:-

Insertion Sort:-

2 7 6a 5 6b

elem < arr[i]

7 < 2 ✗

2 7 6a 5 6b

6a < 7 ✓

6a > 2

2 7 7 5 6b

2 6a 7 5 6b

elem < arr[j]

5 < 7 ✓

2 6a 6a 7 6b

5 < 6a

2 5 6a 7 6b

5 < 2 ✗

elem < arr[j]

6b > 7

2 5 6a 7 7

6b > 6a ✓✓

2 5 6a 6b 7

Stable Sorting algo-

H.W.

a) Bubble sort
Selection sort

I → 2 7 6a 5 6b

II → 2 5 6a 6b 7 D

III → 2 5 6b 6a 7 D