Y section 30 jan

29 January 2025 22:50

for (i=0 ; i<n ; i++){ → O(n).
// statements.
}

(Initialisation, Condition chk, Increment (Up dation).)

for (i=0 ; i<n ; i++) → n times.
{
for (j=0; j<n; j++){ → n times
// statements
}
}

TC - O(n×n) = O(n²)

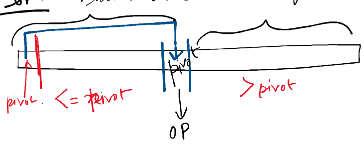
for (i=0 ; i<n ; i++) → n times.
{
for (j=0; j<=1; j++) → 2 times
// statements
}
}

j=0, j=1

O(2×n) ≈ O(n)

j=2, 2<=1 ✗

Quick Sort – Based on divide & conquer technique.



Note –
① Pivot Element.
↓
आप choose karo.
↓
Array का 1st या Last element

35 50 15 25 80 20 90
(p) (q)

⇓

35 20 15 25 80 50 90

pivot = 35
p → pivot का Next element.
q → Arr ka Last element.

p → arr[p] <= pivot.
q ← arr[q] > pivot.

① Chk if p & q have crossed each other
↳ swap (pivot, arr[q]);

① Chk if p & q have not cross each other
↳ swap (arr[p], arr[q])

Algo — 25 20 15 — (35) — 80 50 90 — Algo

<= 35 → left; OP; pivot → right. pivot ind.; > 35

15 20 25 35 50 80 90

25 20 15 pivot = 25

arr(p) <= pivot
arr(q) > pivot.

⇓

(15 20) (25)

<= 25 OP > 25.

Right Array –

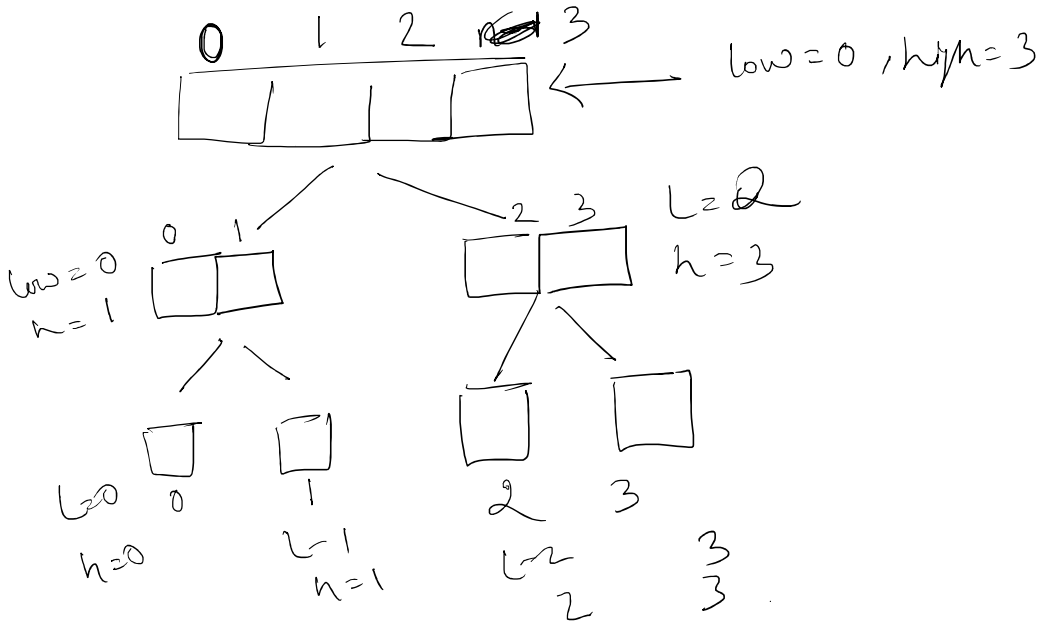
80 50 90
pivot p q
(q) (p)

pivot = 80
p = 50
q = 90

arr(p) <= pivot
arr(q) > pivot.

left arr ⇒ 50 (80) 90 Right

<= 80 OP > 80


0 1 2 3
low = 0 , high = 3
0 1
low = 0
h = 1
2 3
L = 2
h = 3
L = 0
0
h = 0
1
L = 1
h = 1
2
L = 2
3
3
2