# CS & IT ENGINEERING

2024

## Algorithms

### Sorting Algorithms

DPP – 01 Discussion Notes


By- Aditya sir

[MCQ]

Imp

#Q. What is the recurrence relation of insertion sort when the array is almost sorted with p element?

A $T(P) = T\left(\frac{P}{2}\right) + O(1)$ ✗ → $\log_2 P$

B $T(P) = T\left(\frac{P}{2}\right) + P^2$ ✗

C $T(P) = 2T\left(\frac{P}{2}\right) + P$ ✗

D $T(P) = 2T\left(\frac{P}{2}\right) + O(1)$ ✓

Best Case

Insertion Sort :-

Best Case → Time Order → $O(P)$

Worst Case → $O(P^2)$

Ans :- D

Option: A: $T(P) = T(P/2) + O(1)$

$T(P) = T(P/2) + 1$

$T(P/2) = T(P/2^2) + 1$

$T(P) = T(P/2^2) + 2$

$= T(P/2^3) + 3$

$\vdots$

$T(P) = T(P/2^k) + k$

$= T(1) + \log_2 P$

$= O(\log_2 P)$

$\frac{P}{2^k} = 1$

$2^k = P$

$k = \log_2 P$

B) $T(p) = T(p/2) + p^2$ $\quad \underline{O(p)}$

$=$

$= a=1,\ b=2,\ f(p) = p^2$

$\log_b^a = 0$

$\Rightarrow \underline{\Omega(p^2)}$

C) $T(p) = \underline{2T(p/2)} + p \longrightarrow$ Best Case

$T(p/2) = \underline{2T(p/2^2)} + p/2$

$T(p) = 2\left[2T(p/2^2) + p/2\right] + p$

$= 2^2 T(p/2^2) + 2 \times \frac{p}{2} + p$

$$T(P) = 2^2 T(P/2^2) + P + P$$

$$= 2^2 T(P/2^2) + 2P$$

$$= 2^3 T(P/2^3) + 3P$$

General Term :

$$T(P) = 2^K T(P/2^K) + K * P$$

For B.C, $P/2^K = 1$

$$2^k = p \implies \boxed{k = \log_2 p}$$

$$T(p) = p * T(1) + p * \log_2 p$$

$$= p * c + p * \log_2 p$$

$$\boxed{T(p) = O(p \log_2 p)}$$

D) $T(P) = 2T(P/2) + 1$ → $O(1)$

$T(P/2) = 2T(P/2^2) + 1$

$T(P) = 2\left[2T(P/2^2) + 1\right] + 1$

$= 2^2 T(P/2^2) + 2 + 1$

$= 2^3 T(P/2^3) + 2^2 + 2^1 + 2^0 \ldots$

General term

$$T(p) = 2^k T(p/2^k) + (2^0 + 2^1 \cdots 2^{k-1})$$

GP $\Rightarrow$ $a = 1$, $r = 2$, $n = k$ $\Rightarrow$

$$\text{Sum} = \frac{a(r^n - 1)}{r - 1}$$

$$= \frac{1 \times (2^k - 1)}{2 - 1}$$

$$= (2^k - 1)$$

General Term

$$T(P) = 2^K T(P/2^K) + (2^K - 1)$$

For B.C, $P/2^K = 1$ $\quad 2^K = P$

$$K = \log_2(P)$$

$$T(P) = P * T(1) + (P - 1)$$

$$= P * C + P - 1$$

$$= \boxed{O(P)}$$

[NAT]

#Q. Consider the following A with 8 elements:


60

What is the index value of elements after 3rd pass of selection sort?

Ans:- 6

I/p: 70 60 20 50 40 5 19 21

pass1 :- o/p: 5 | 60 20 50 40 70 19 21

pass2 o/p: 5 19 | 20 50 40 70 60 21

pass 3 o/p: 5 19 20 | 50 40 70 60 21

pass 4 o/p: 5 19 20 21 | 40 70 60 50

Pass 5 o/p: 5 19 20 21 40 | 70 60 50

pass 6 o/p: 5 19 20 21 40 50 | 60 70

pass 7 o/p: 5 19 20 21 40 50 60 70

↳ (n-1) passes always.

[MCQ]

#Q. Consider the following array with 8 elements

| 50 | 60 | 90 | 65 | 55 | 45 | 85 | 12 |
|---|---|---|---|---|---|---|---|

What is result after 3rd pass of bubble sort?

A 50, 60, 55,45, 12, 65, 85, 90

B 12, 45, 50, 60, 90, 65, 55 85

C 90, 85, 65, 50, 60, 55, 45, 12

D 50, 55, 45, 60, 12, 65, 85, 90

Ans: D

3rd pass → max last 3 elem sorted

given: 50 60 90 65 55 45 85 12

↓

pass 1 :- o/p: 50 60 65 55 45 85 12 (90)

pass 2 o/p: 50 60 55 45 65 12 (85) (90)

⇒ pass 3 o/p: 50 55 45 60 12 (65) (85) (90)

↓

pass 4 o/p: 50 45 55 12 (60) (65) (85) (90)

pass 5 o/p: 45 50 12 (55) (60) (65) (85) (90)

|

pass 6 o/p: 45 12 50 55 60 65 85 90

pass 7 o/p: 12 45 50 55 60 65 85 90

Sorted

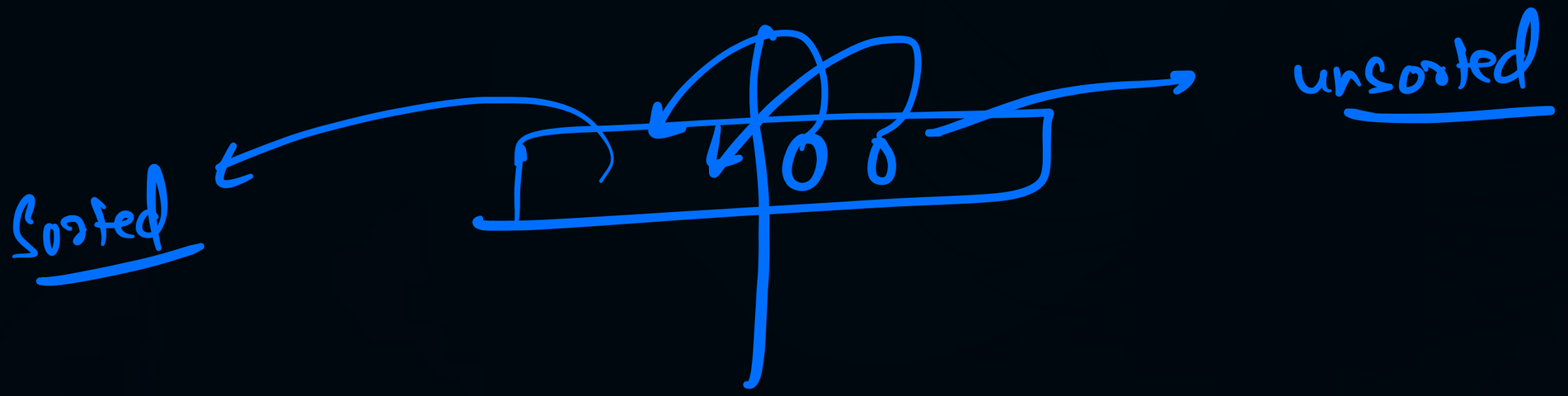
(n-1) passes

[NAT] → Numerical

Ans: 16

#Q. Consider the following array A with 9 elements


How many swaps are needed to sort the array by using insertion sort___?



Given Input :- 10 20 50 60 70 65 55 25 15

pass 1 : 10 | 20 50 60 70 65 55 25 15
→ 0 swaps

pass 2 : 10 20 | 50 60 70 65 55 25 15
→ 0 swaps

pass 3 : 10 20 50 | 60 70 65 55 25 15
→ 0 swaps

pass 4 : 10 20 50 60 | 70 65 55 25 15
→ 0 swaps

pass 5 : 10 20 50 60 70 | 65 55 25 15
→ 1 swaps

pass 6: 10 20 50 60 65 70 | 55 25 15 → 3 swaps

pass 7: 10 20 50 55 60 65 70 | 25 15 → 5 swaps

pass 8: 10 20 25 50 55 60 65 70 | 15 → 7 swaps

O/p: 10 15 20 25 50 55 60 65 70 → Sorted.

Total Swaps: 0 + 1 + 3 + 5 + 7

= 16 swaps

[MSQ]

#Q. Bubble sort is-

Ans :- A, D

A In place sorting technique ✓ → O(1) space

B Outplace sorting technique ✗ → not in place

C Unstable sorting technique ✗

D Stable sorting technique ✓

Stable & Inplace.

[MCQ]

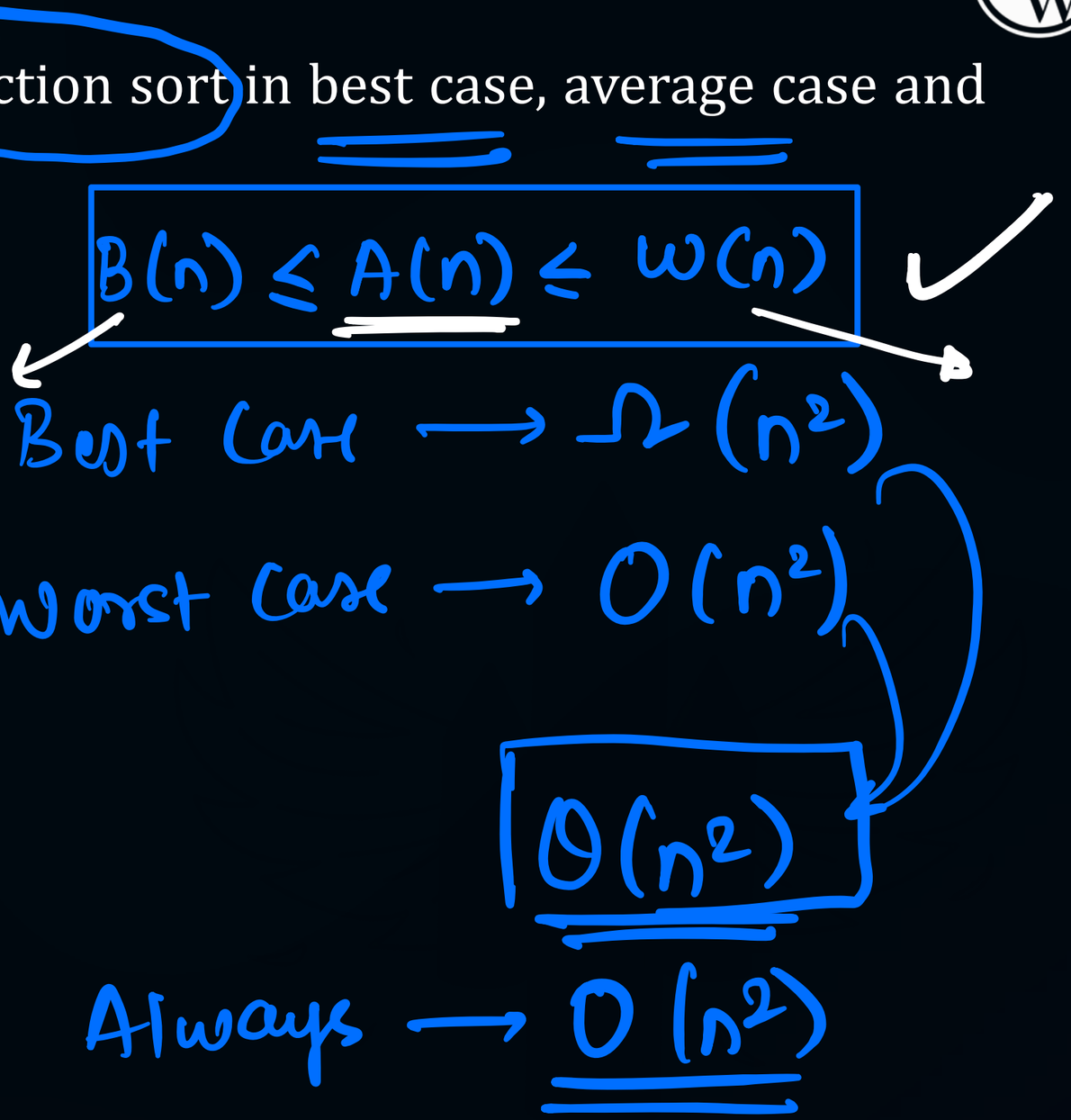
#Q. What is the time complexity of selection sort in best case, average case and worst case respectively is:

A. $O(n), O(n\log n), O(n^2)$

B. $O(n), O(n^2), O(n^2)$

C. $O(n^2), O(n^2), O(n^2)$

D. $O(n), O(n\log n), O(n\log n)$



[MCQ]

#Q. How many swaps and needed in selection sort to sort n element in worst case?

↳ takes min swap among all comp algos in use.

Ans: A

A. n − 1

B. $\frac{n(n-1)}{2}$ ✗

C. n ✗

D. $n^2$ ✗

Selection Sort ⇒ $\Theta(n^2)$ → Comp + Swaps

Always (n-1)

[MCQ]

#Q. Consider the following elements:

| 101 | 56 | 934 | 555 | 8 | 12 | 785 | 23 | 5 | 999 |
|---|---|---|---|---|---|---|---|---|---|

What is the result after 3rd pas of radix sort?

A. 56, 23, 12, 8, 5, 101, 555, 785, 934, 999

B. 56, 23, 12, 8, 5, 101, 55, 785, 999, 934

C. 5, 8, 12, 23, 56, 101, 555, 785, 934, 999

D. 5, 8, 12, 23, 56, 101, 555, 785, 934, 999

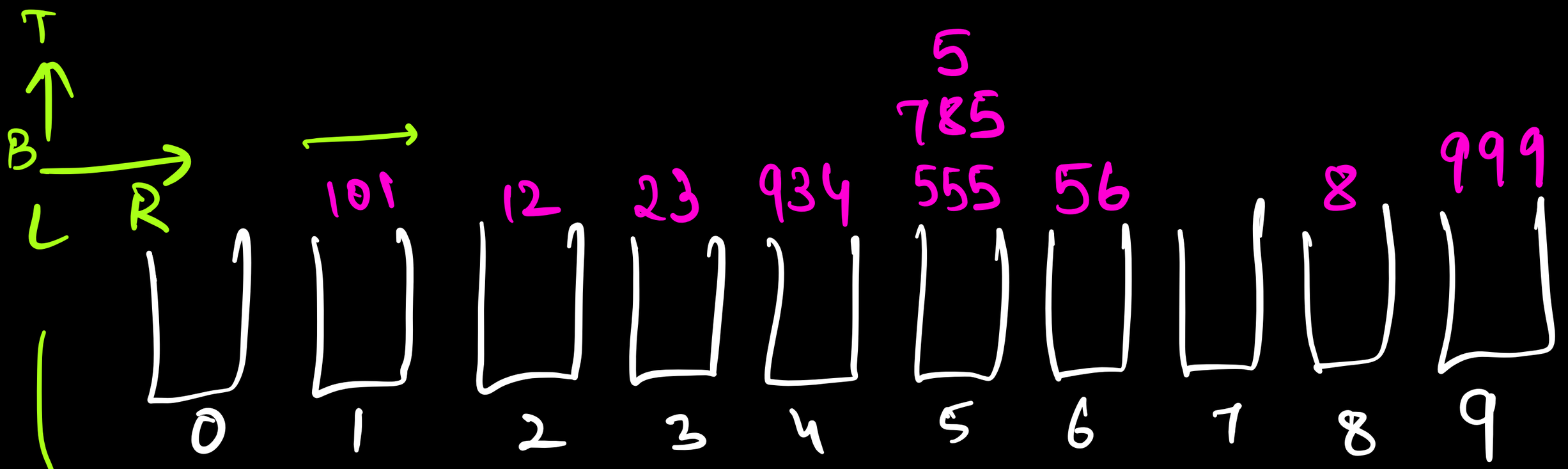
radix Sort
↳ non-comparison
↓
buckets

Ans: D

A: 101 56 934 555 8 12 785 23 5 999

→ 1 6 4 5 8 2 5 3 5 9

Pass1 :- Bucket as per last digit (1st digit from Right)




O/p: Pass1: 101, 12, 23, 934, 555, 785, 5, 56, 8, 999

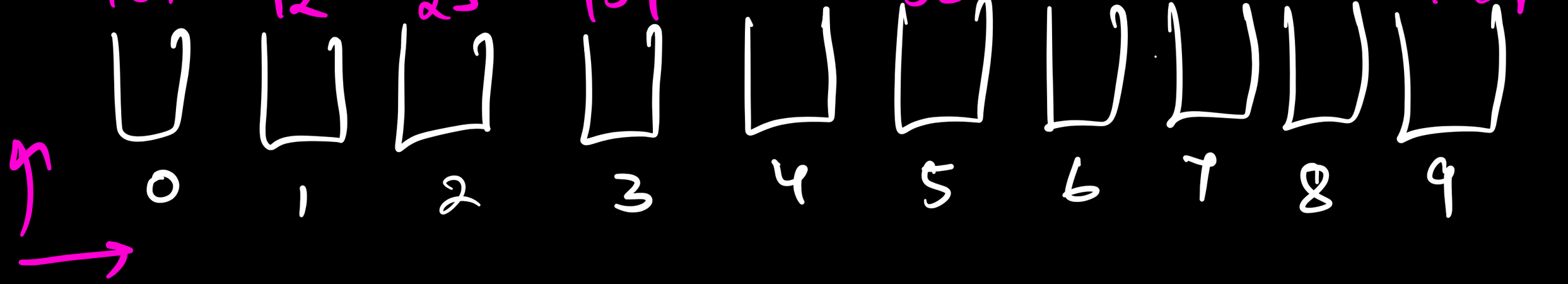
$L_0$ $L_1$ $L_2$ $L_3$ $L_5$ $L_8$ $L_0$ $L_5$ $L_9$

pass 2 : Bucket as per 2nd last digit (2nd digit from right side.

5 → 05 → (0)

8 → 08

8
5
101 12 23 934 555 56 785 999




pass 2 o/p: 101, 5, 8, 12, 23, 934, 555, 56, 785, 999

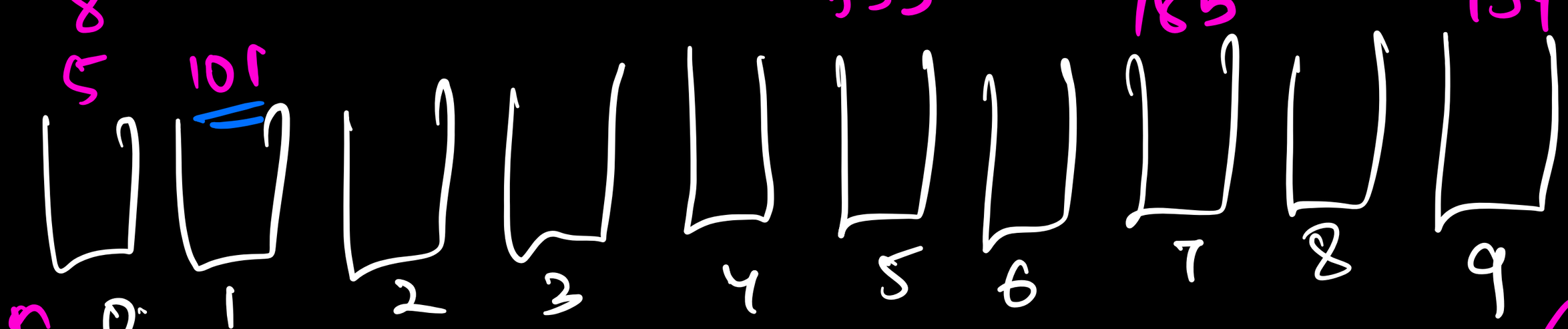
1 0 0

Pass 3: Bucket as per 3rd digit from right side.
(100's place digit)

56
23
12
8

5 → 005
12 → 012

555
785
999
934




→ Pass 3 O/p: → Sorted

5, 8, 12, 23, 56, 101, 555, 785, 934, 999

999

# THANK - YOU

→ Lectures
→ DPP/WT
→ Practice
PYQs