Data Science & Artificial Intelligence

Algorithms

Test Series 1500+

Lecture - 04 05

By- Aditya sir

Recap of Previous Lecture

Topic

Topic

Time Complexity
Sorting
Graphs
Misc

Questions

Topics to be Covered

Topic

Topic

Sorting

Graphs

Misc → Algo

Questions

[NAT]

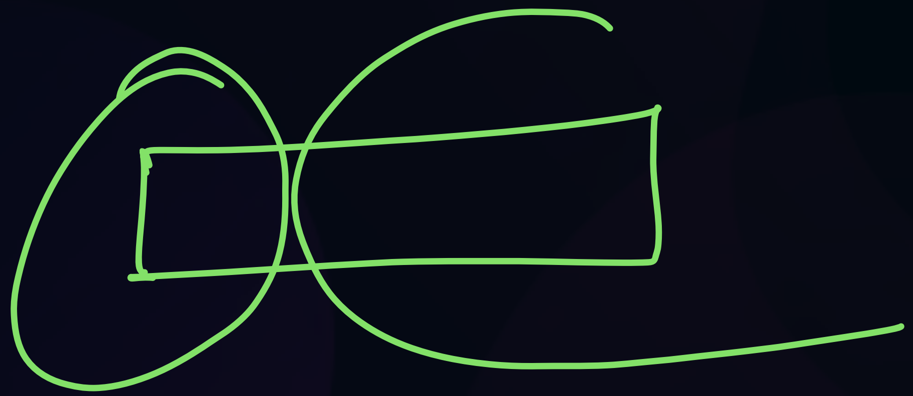
#Q5. Consider the following array

A	70	60	23	28	55	81	116

If Insertion sort is applied to sort in ascending order and If number of swaps are X and number of comparisons are Y then the value of X*Y is_____for above array A?

Swaps → X

Comparisons → Y



Soln: 70 60 23 28 55 81 116.

S | C

Pass1: 70 | 60 23 28 55 81 116 → 1 | 1 ✓

Pass2: 60 70 | 23 28 55 81 116 → 2 | 2 ✓

Pass 3: 23 60 70 | 28 55 81 116 → 2 | 3 ✓

Pass 4: 23 28 60 70 | 55 81 116 → 2 | 3 ✓

Pass 5: 23 28 55 60 70 | 81 116 → 0 | 1 ✓

Pass 6: 23 28 55 60 70 81 | 116 → 0 | 1 ✓

7 | 11

→ sorted

$X = 7$

$Y = 11$

Ans: $X * Y = \underline{\underline{77}}$

Topic : Divide and Conquer

#Q3. Let's suppose you are given an array of n elements in which, the few elements in the beginning are $ and remaining elements are @, then what is the complexity of most efficient algorithm to find the first @ symbol?

A $\theta(n \, logn)$

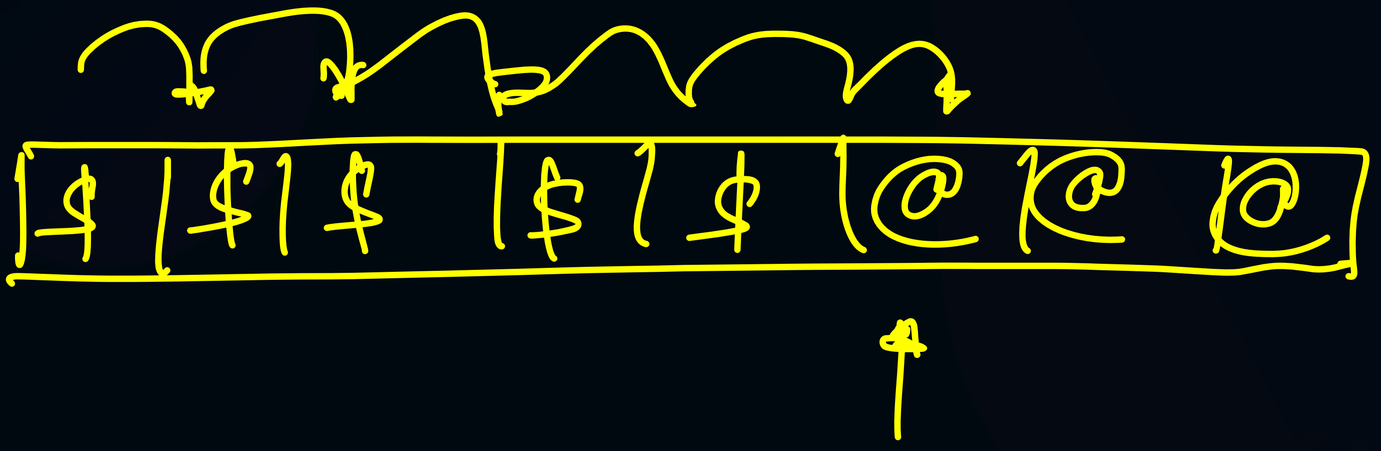
B $\theta(n)$ → LS

C $O(1)$

[✓ D] $\theta(\log n)$ → BS

Ans: D

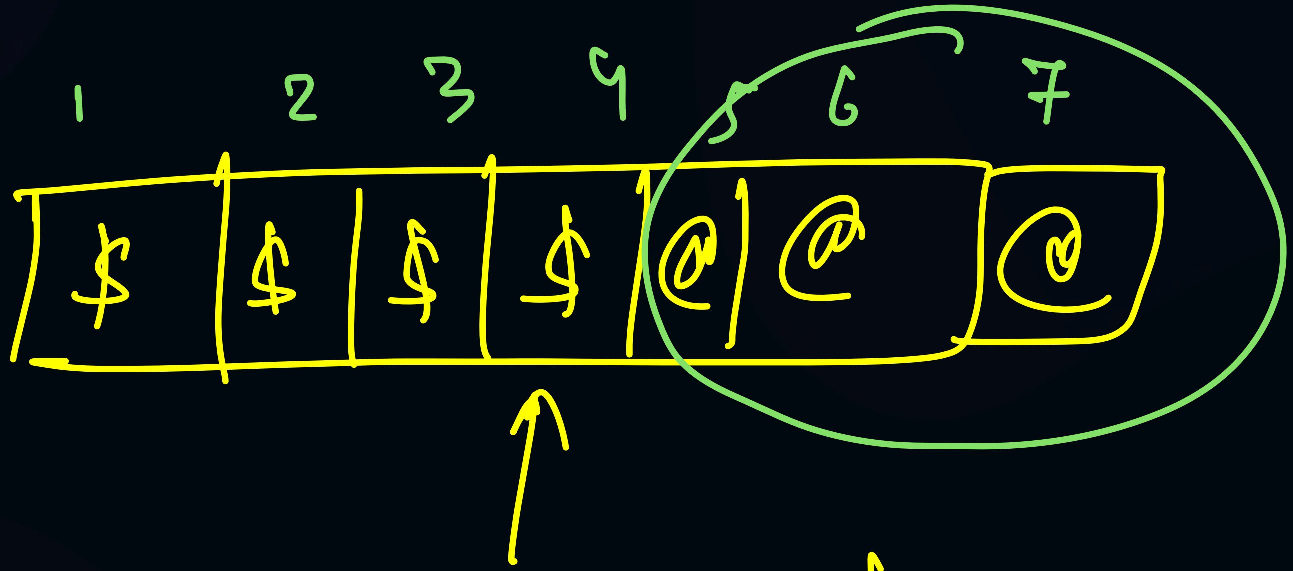
Soln:



$	$	$	$	$	@	@	@

Appr1: Linear Search : WC: $O(n)$

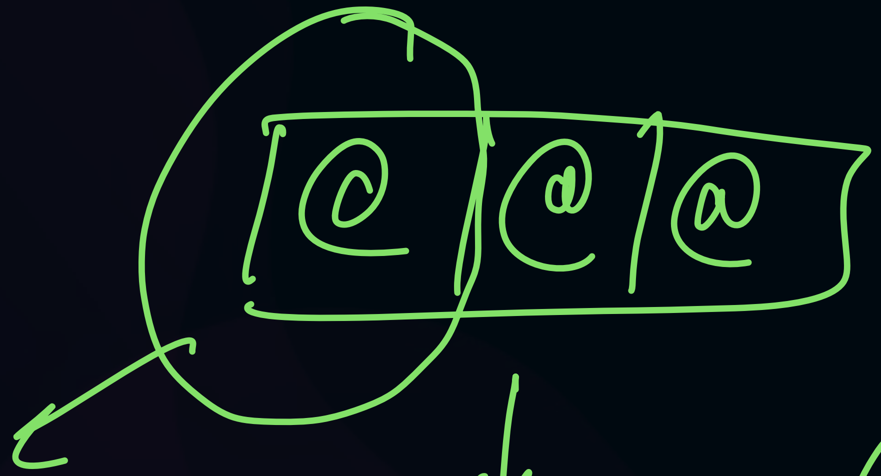
Appr2: Array is sorted → Binary Search



mid = $

ans = 6

ans = 5



mid = @

[MCQ]

#Q4. Consider the following statements:

✗ S_1: Number inversions are same as number of comparisons in ~~merge~~ Selection sort.

✗ S_2: Number of inversions are same as number of ~~swaps~~ Comparisons in insertion sort.

Which of the following is correct.?

A S_1 only ✗

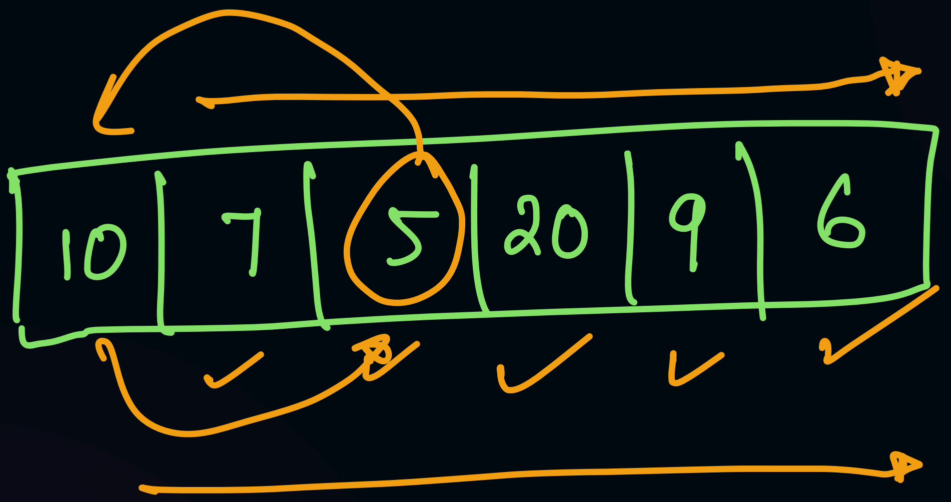
B S_2 Only ✗

C Both ✗

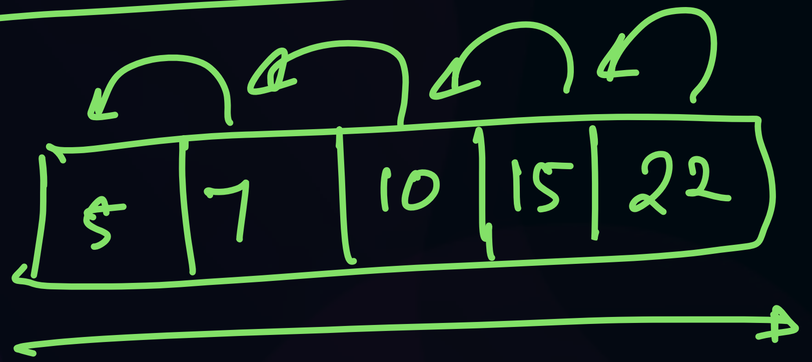
[✓ D] None

Ans: D

Soln: S1: Selection Sort



S2: Insertion Sort:



4 comparisons

(n-1) Comp

inversions = 0 ⟷

Topic : Divide and Conquer

#Q9. Generally, merge sort is a divide and conquer technique which can also be implemented in a recursive manner. If there are 300 elements in an array then how many recursive functions calls are needed in merge sort?

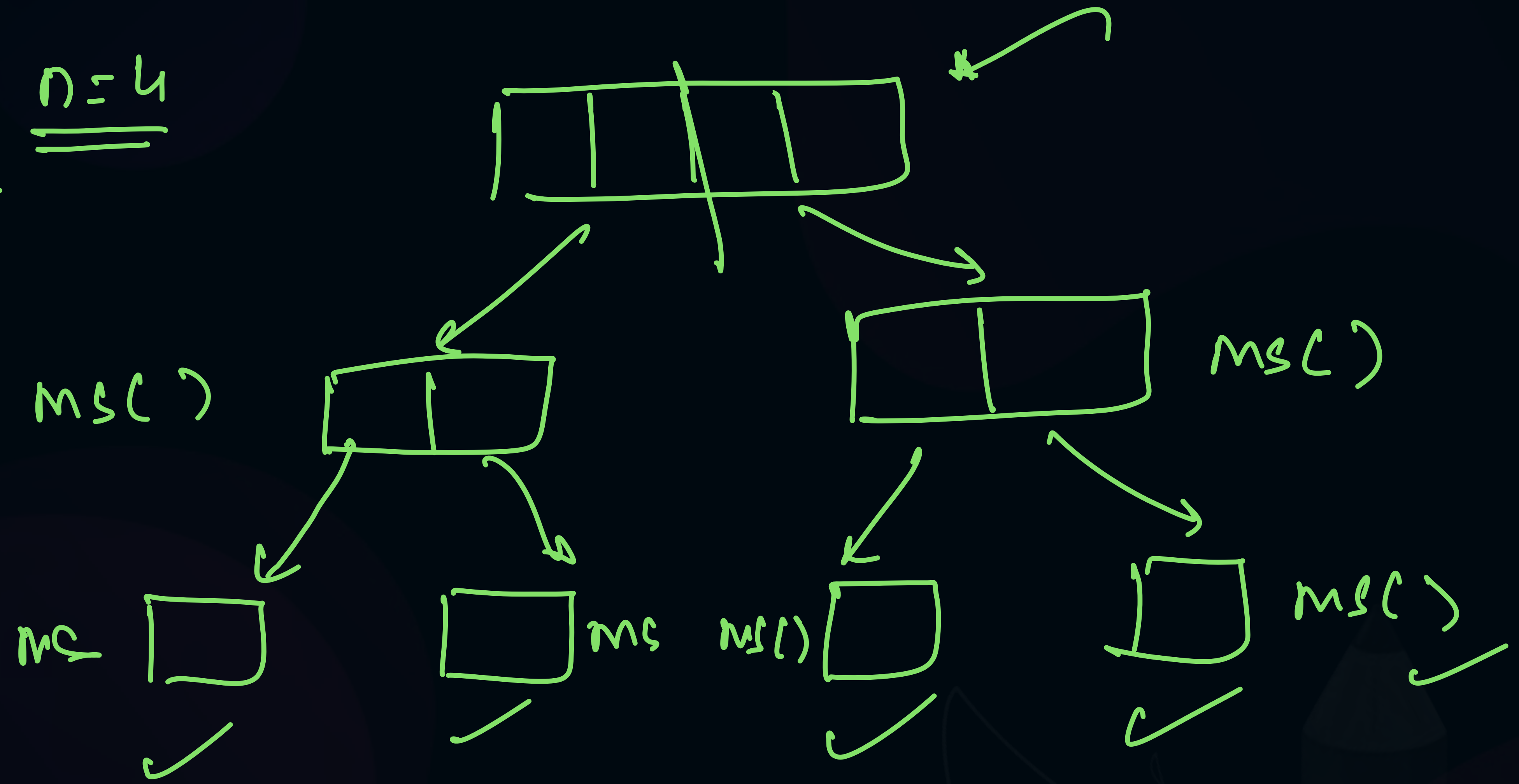
20 %

Ans:- 600 - 2

= 598

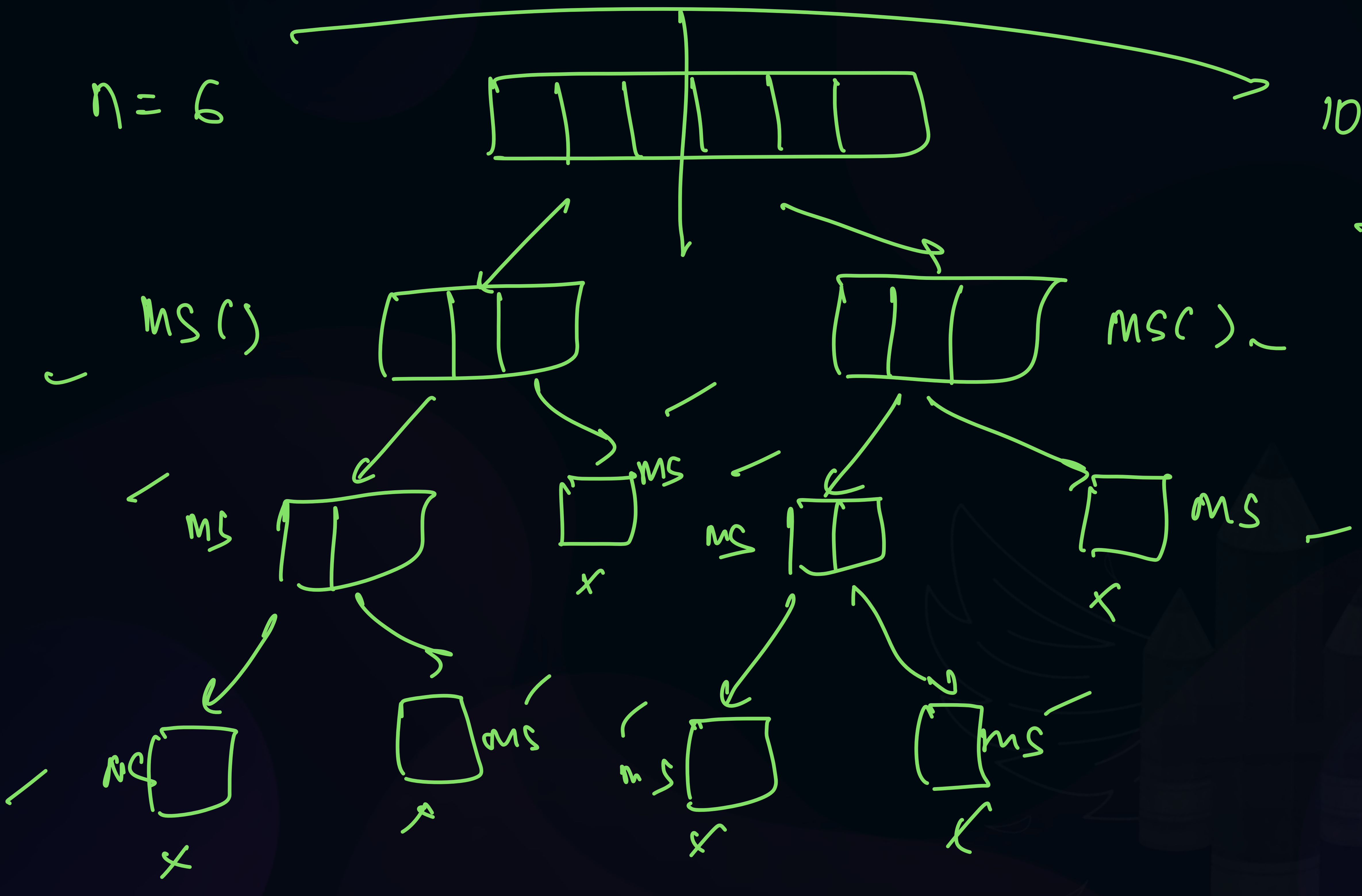
Soln :

n = 4



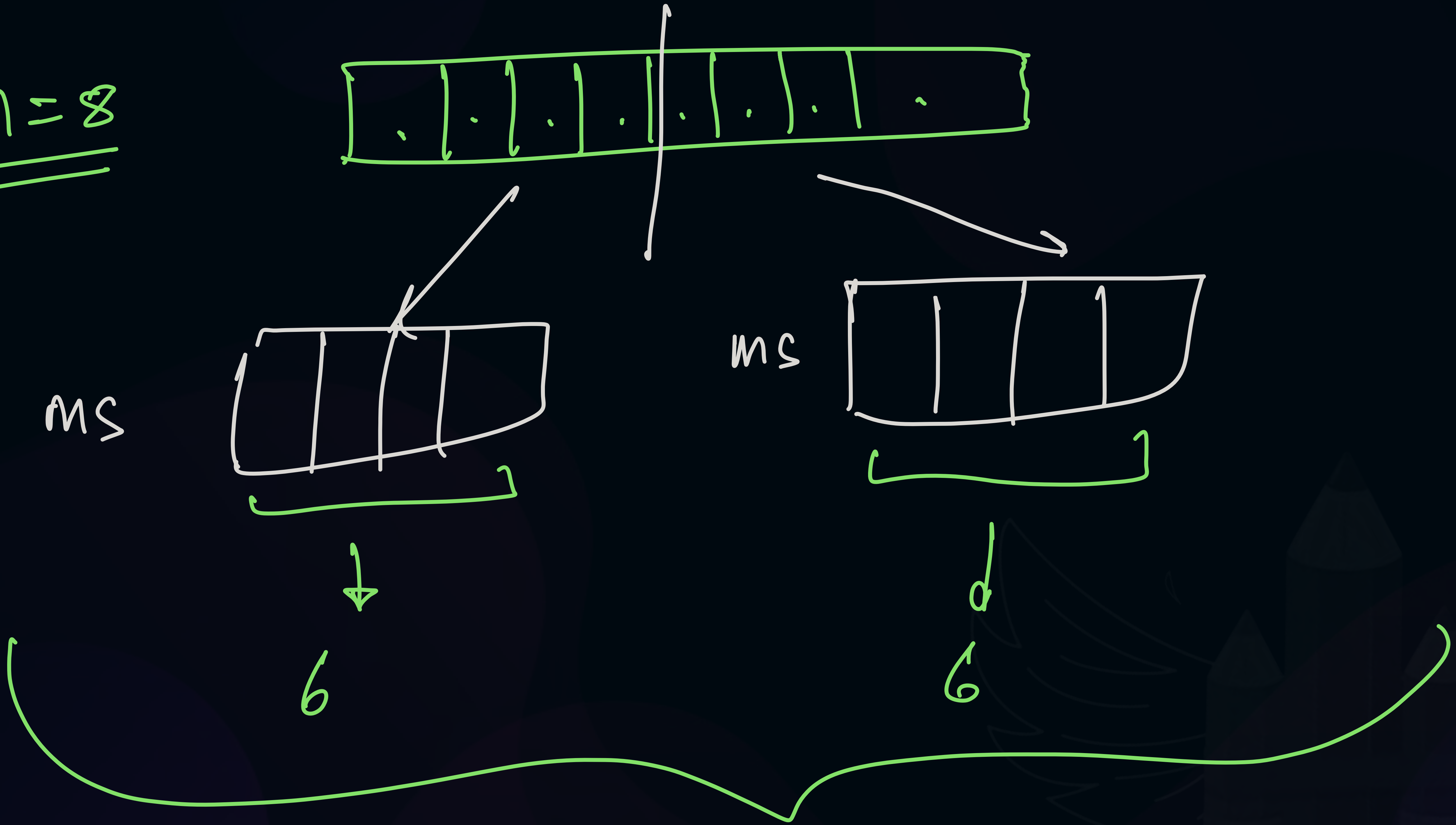
6 Recursion calls

$n = 6$



10 Rec calls

$n = 8$



$6 + 6 + 2 = $ 14 calls

n size array ⟶ $\underline{\underline{2n-2}}$ Recursive calls

For $\underline{n=300}$, $2\times300-2$

$= 600-2$

$= \boxed{598}$ Recursive calls

Topic : Divide and Conquer

#Q8. What is the auxiliary space complexity of merge sort

not inplace

A O(1)

B O(n)

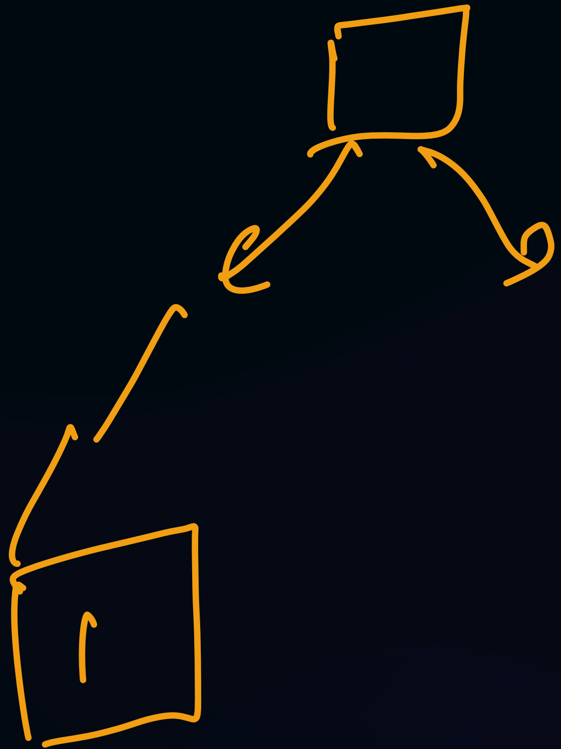
C O(logn)

D ~~[illegible]~~ O(nlogn)

Ans: B

WC.

Soln: Merge Sort : Space Complexity



① Temporary Array (B) used for merging $\Longrightarrow O(n)$

② Recursion Stack → $O(\log_2 n)$

Overall SC : $O(n + \log_2 n) = O(n)$

[MSQ]

#Q8. Which of the following sorting technique is/are stable sorting technique.

A Bubble sort ✓

B Insertion sort ✓

C Selection sort ✗

D Quick sort ✗

Stable

1. Bubblo sort
2. Insertion Sort
3. Merge Sort
4. Radix Sort

Un-stable

1. Selection sort
2. Heap sort
3. Quick Sort.

Ans: A, B

Inplace

1. Bubble sort
2. Selection Sort
3. Insertion Sort
4. Quick Sort
5. Heap Sort

Not inplace

1. Merge Sort
2. Radix Sort

Topic : Heaps & Sorting

#Q.1 Which of the following is valid max heap?

A 28, 27, 12, 23, 22, 6, 5

B 190,100,10,90,80, 9,1, 70

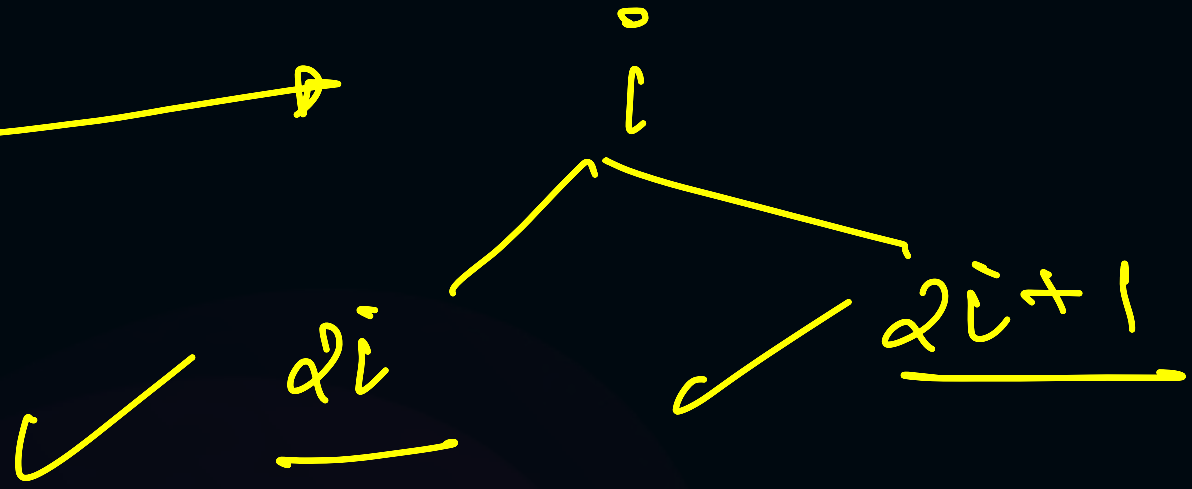
[✓ C] Both (a) and (b)

✗ D Neither (a) nor (b)

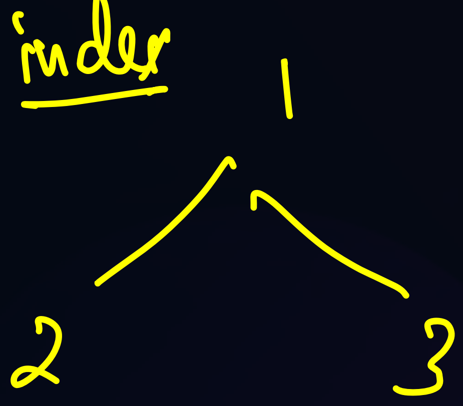
Ans : C

Sol^n :- Array Representation of a max-Heap

Imp →



(starting at 1)



Child ⇒ K

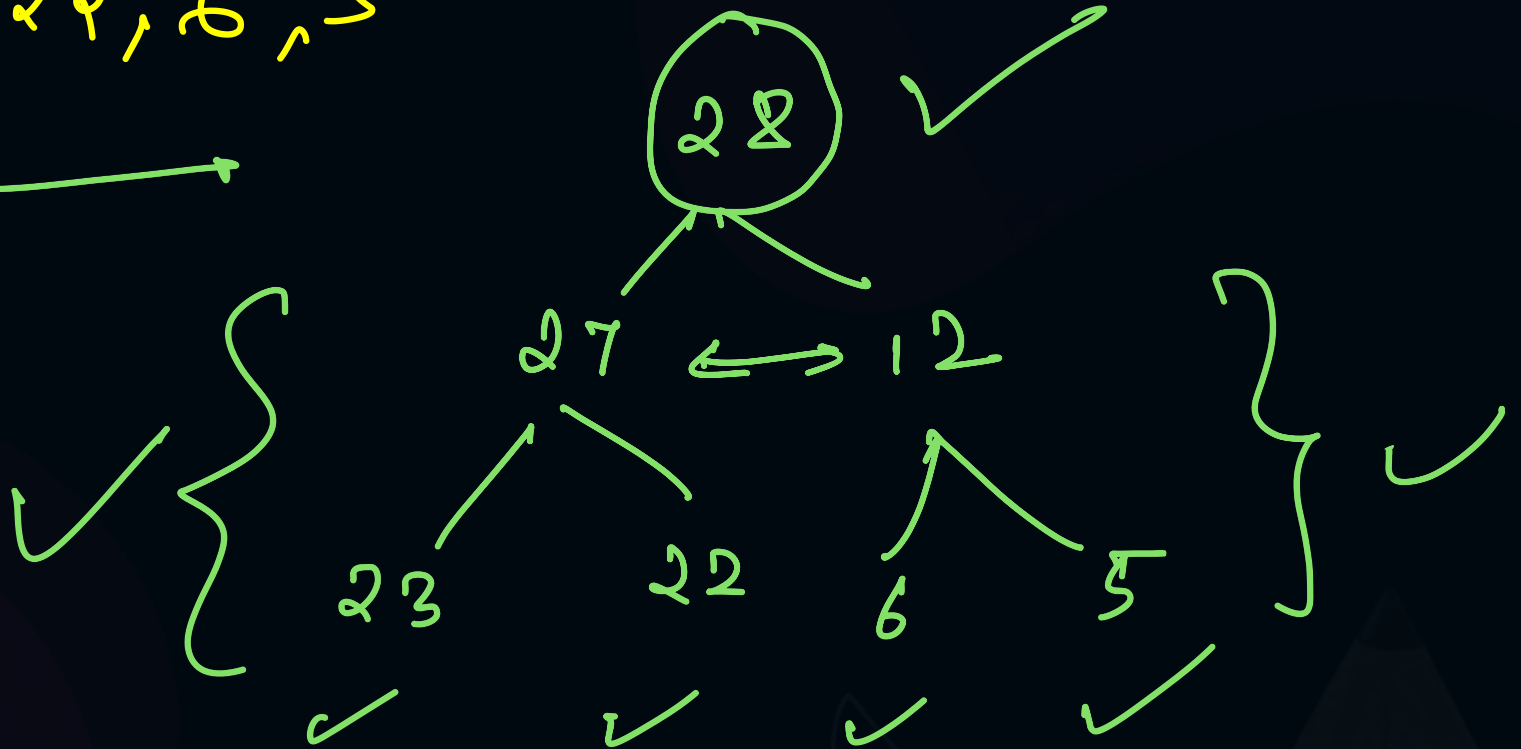
parent ⇒ $\lfloor K/2 \rfloor$ ✓

$3 \rightarrow \lfloor 3/2 \rfloor = \lfloor 1.5 \rfloor = \boxed{1}$

$\lfloor 2/2 \rfloor = 1$

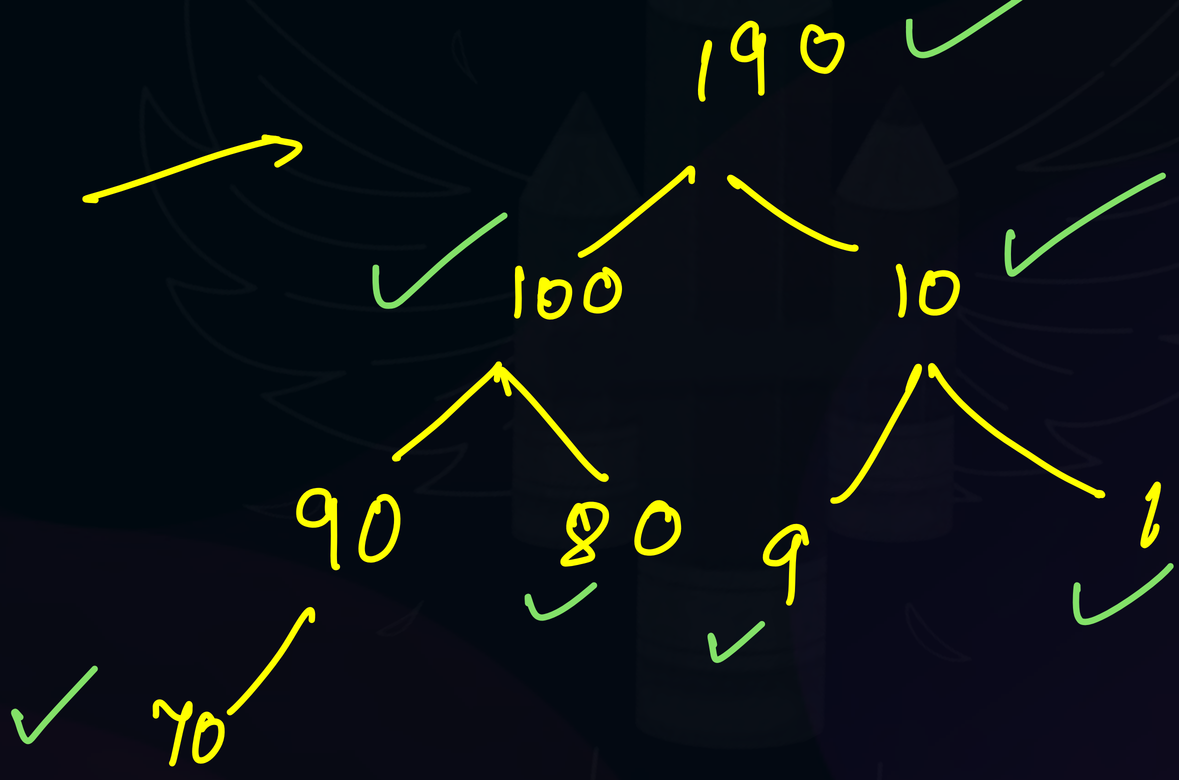
A) 28, 27, 12, 23, 22, 6, 5

valid max-Heap



B) 190, 100, 10, 90, 80, 9, 1, 70

valid max-Heap



[NAT]

#Q6. Consider the following array

A	500	700	130	90	850	210	160

How many comparisons are needed to ~~start~~ Sort the above array using insertion sort?

50%

Total Comparisons = 16

Soln :- A: 500 700 130 90 850 210 160

		Comp
pass1:	500 | 700 130 90 850 210 160	→ 1
↳	500 700 | 130 90 850 210 160	→ 2
pass2: ↳	130 500 700 | 90 850 210 160	→ 3
pass3: ↳	90 130 500 700 | 850 210 160	→ 1
pass4: ↳	90 130 500 700 850 | 210 160	→ 4
pass5: →	90 130 210 500 700 850 | 160	→ 5
pass6: →	90 130 160 210 500 700 850 ||	16

[MCQ]

#Q.3 Consider a complete binary tree where the left and right subtrees of the root are max -heaps. The lower bounds for the number of operations to convert the tree to a heap is

A $\Omega(\log n)$

[✓ B $\Omega(1)$]

C $\Omega(n \log n)$

D $\Omega(n^2)$

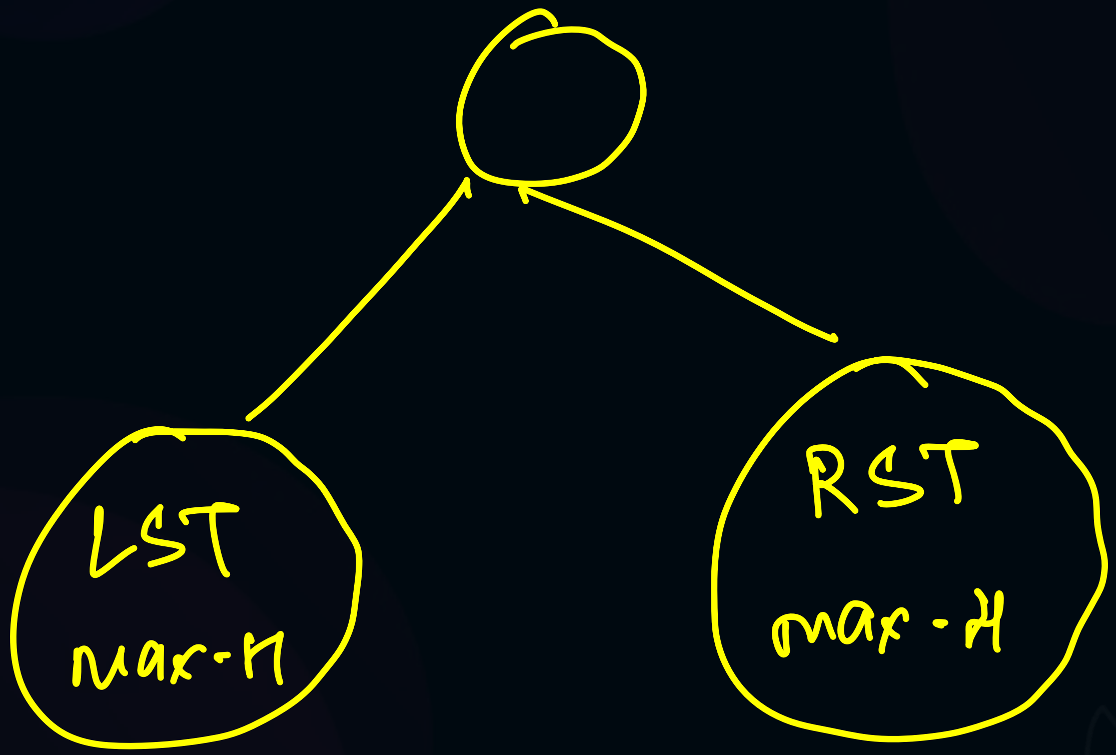
Ans : B

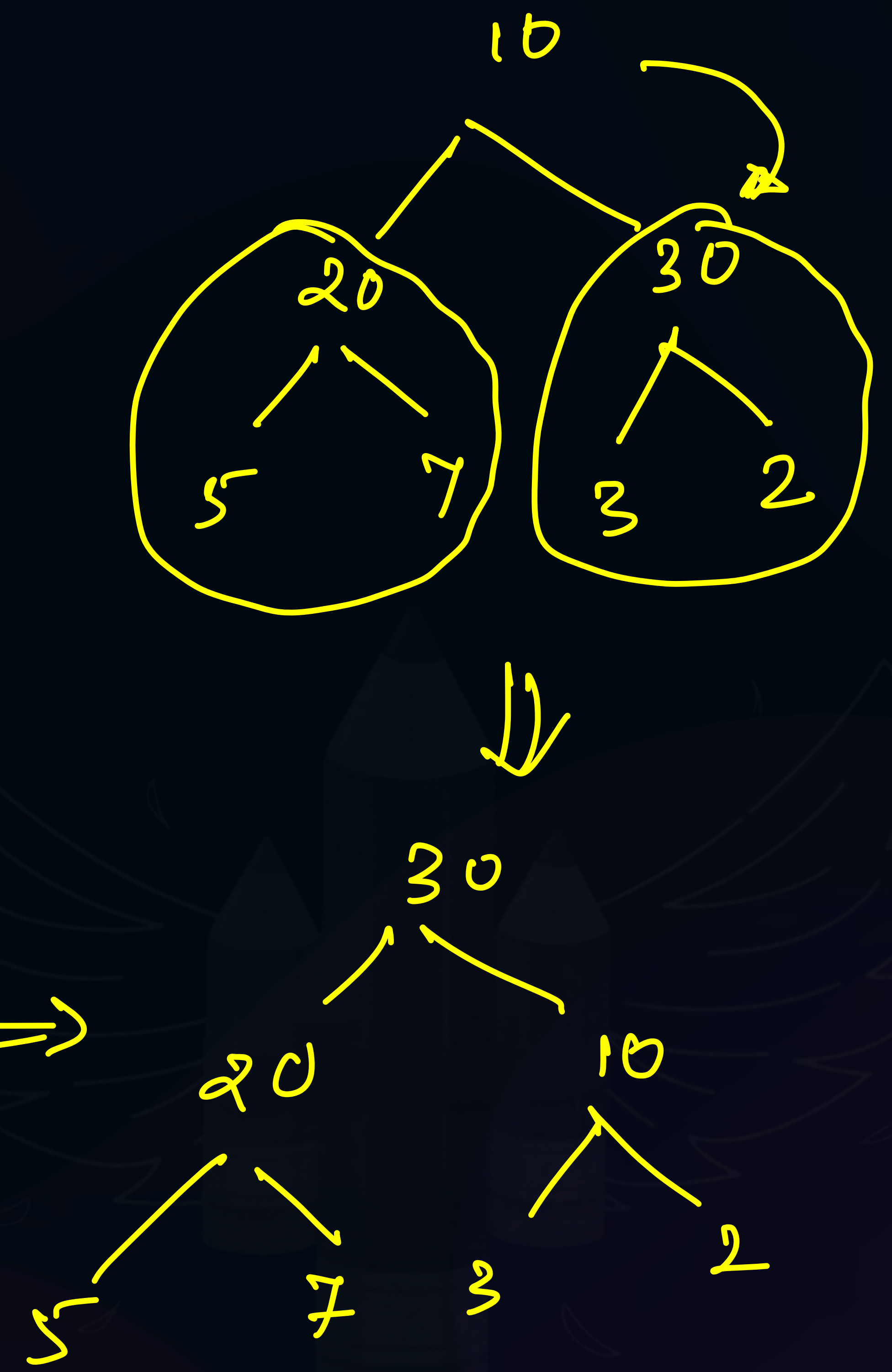
Lower Bound → $\Omega(1)$

Upper Bound → $O(\log_2 n)$

33%

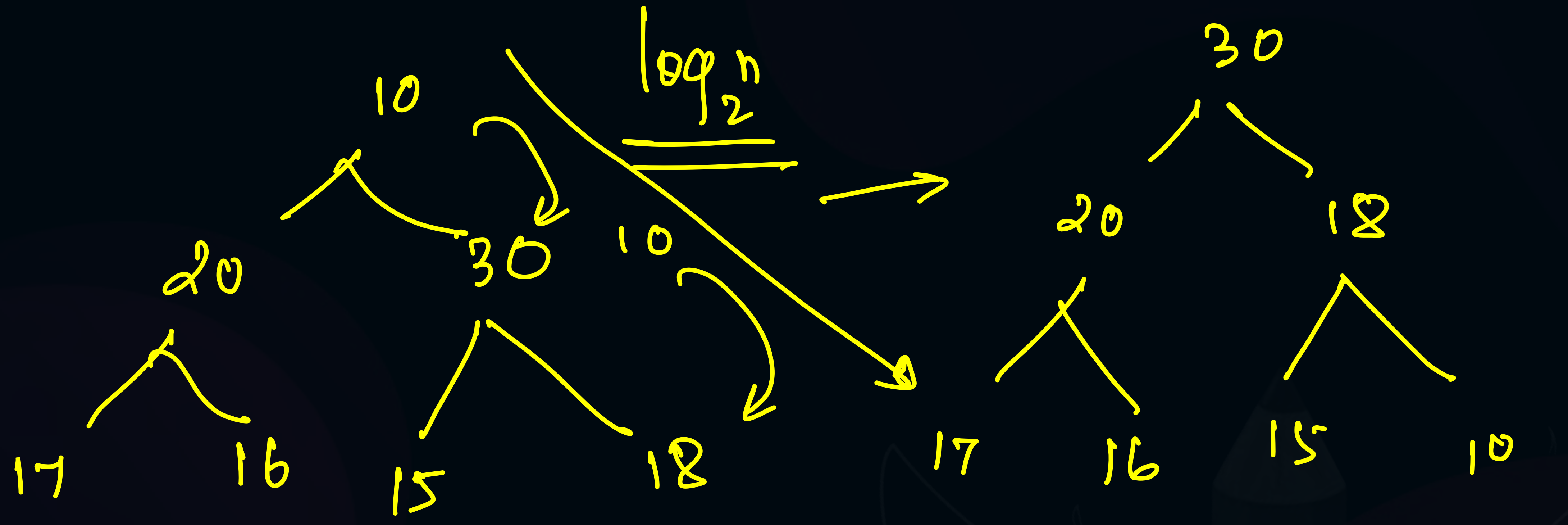
Soln:-





Lower Bound ⟹

for Worst Case → Upper Bound



max - Heap

[MCQ]

#Q.4 Consider a max heap represented by the array :

~~50, 40, 30, 20, 10, 15, 16, 17, 8, 4~~

Array	1	2	3	4	5	6	7	8	9
Value	50	40	30	20	25	26	27	18	14

Now consider that a value 35 is inserted into the heap. After insertion new heap is : -

A 50, 40, 30,20,25,26,27, 18, 14, 35

B 50, 40, 30,20, 26,35,27, 18,14,25

C 50, 40, 30,20, 35,26,27, 18,14,25

D 50, 40, 30,20, 35,26,27, 14,18,25

Ans : C

Soln :-

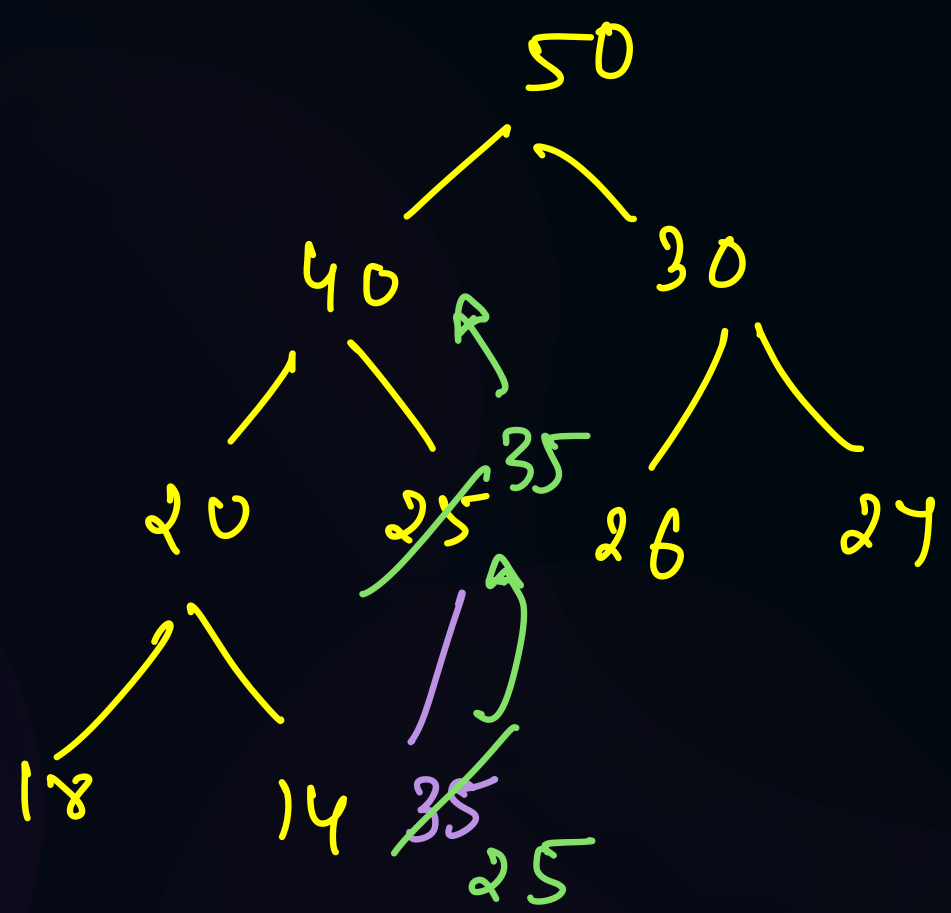
given : 50 40 30 20 25 26 27 18 14

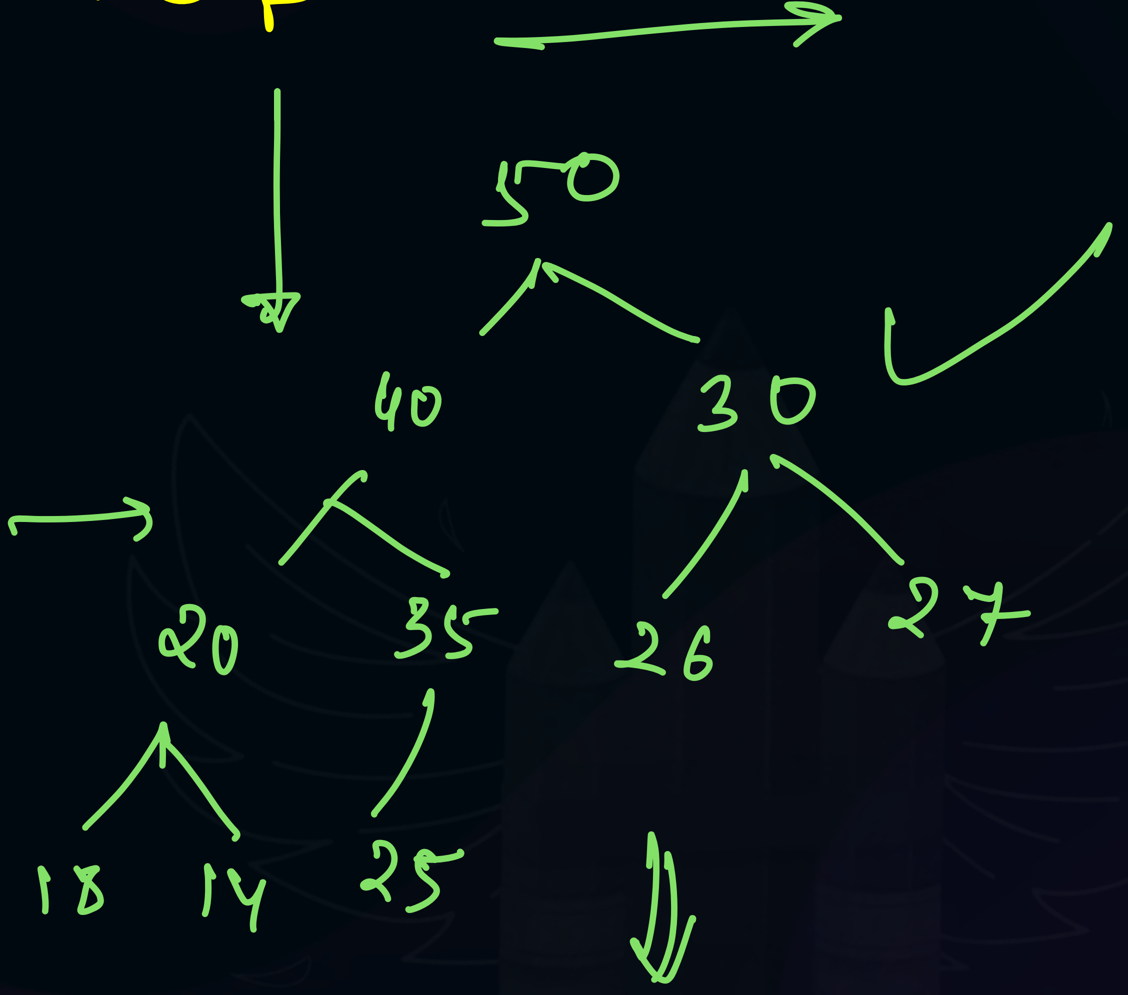
↓ max - Heap

Insertion operatn

↓

Bottom - top





50 40 30 20 35 26 27 18 14 25

Topic : Divide and Conquer

#Q12. Assume that there are 8n sorted list of size n/4 then what is the time complexity of merging them into single sorted list

A $\theta(n^2logn)$ ✓

B θ(log n) ✗

C θ(nlogn) ✗

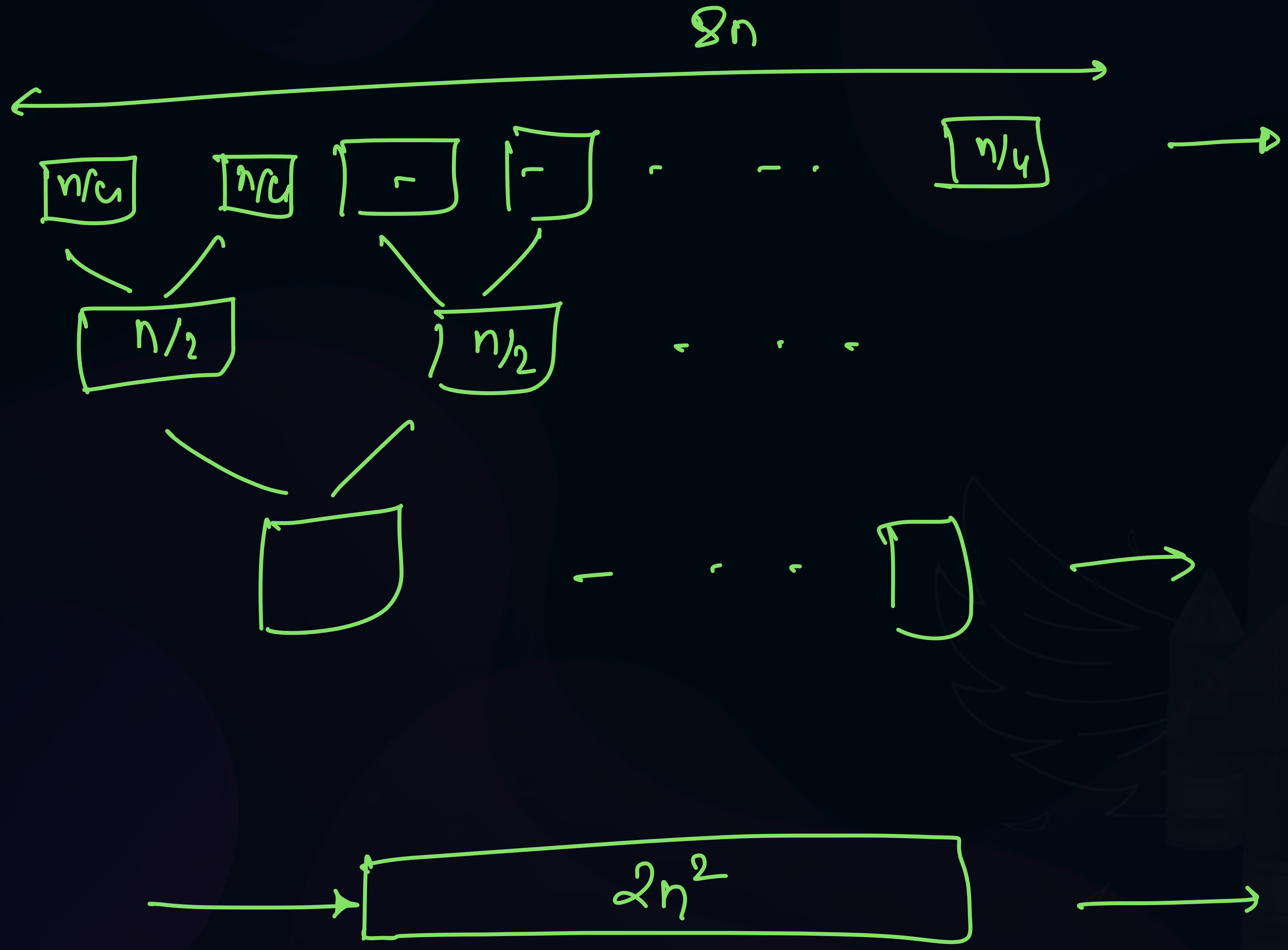
D $\theta(n^2)$ ✗

Merging Algo

Ans: A

default → 2 way merging

Soln:- 8n Sorted lists ⟶ n/4



8n lists

↓

$\frac{8n}{2} = 4n$ lists

$\frac{4n}{2} = 2n$ lists

⋮

1 list

No. of levels $\Longrightarrow$ $8n \xrightarrow{\frac{1}{2}} 4n \xrightarrow{\frac{1}{2}} 2n \xrightarrow{\frac{1}{2}} n \cdots 1$

$\hookrightarrow \log_2(8n) = \log_2 8 + \log_2 n$

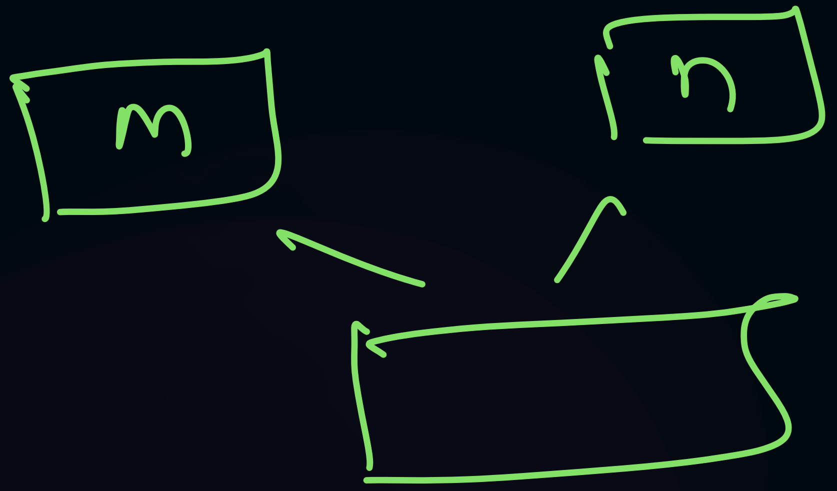
$= \underline{\underline{(3 + \log_2 n)}} = O(\log_2 n)$

No. of elems at every level

$= 8n * n/4 = \boxed{2n^2}$ elems

Merging $2n^2$ elems $\Longrightarrow O(2n^2)$

$= \underline{\underline{O(n^2)}}$



$\longrightarrow O(m+n)$ ops

every level $\longrightarrow$ TC: $O(n^2)$

no. of levels $\longrightarrow O(\log_2 n)$

Here Total TC

= No of levels * TC of each level

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

$= \underline{\underline{O(n^2 \log n)}}$

[MCQ]

#Q9. Consider the following array with 8 elements:

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

D 50, 55, 45, 60, 12, 65, 85, 90 ✗

1st → 1

2nd → 21

3rd → 321

Ans : C

Soln :-

i/p : 60 50 45 85 55 90 65 12

pass1 : ~~60~~ ~~50~~ ~~45~~ ~~85~~ ~~55~~ ~~90~~ ~~65~~ ~~12~~

50 ~~60~~ 60 55 85 65 ~~90~~ 90

45 12

pass2 : ~~50~~ ~~45~~ ~~60~~ ~~55~~ ~~85~~ 65 ~~12~~ 90

45 50 55 60 65 ~~85~~ 85

12

pass3 : 45 50 55 60 ~~65~~ ~~12~~ 85 90 O/p of pass 2

12 65

pass4 45 50 55 60 12 65 85 90

pass 5 : 45 50 55 12 60 65 85 90

$\rightarrow$ 45 50 12 55 60 65 85 90

pass 6

pass 7 45 12 50 55 60 65 85 90

6 $\rightarrow$ 12 45 50 55 60 65 85 90

$\boxed{n-1}$

2 mins Summary

Topic

Sorting

Topic

Heaps

Topic

Tc + Dnc

Misc

Topic

Questions

THANK - YOU

Telegram Link for Aditya Jain sir:
https://t.me/AdityaSir_PW