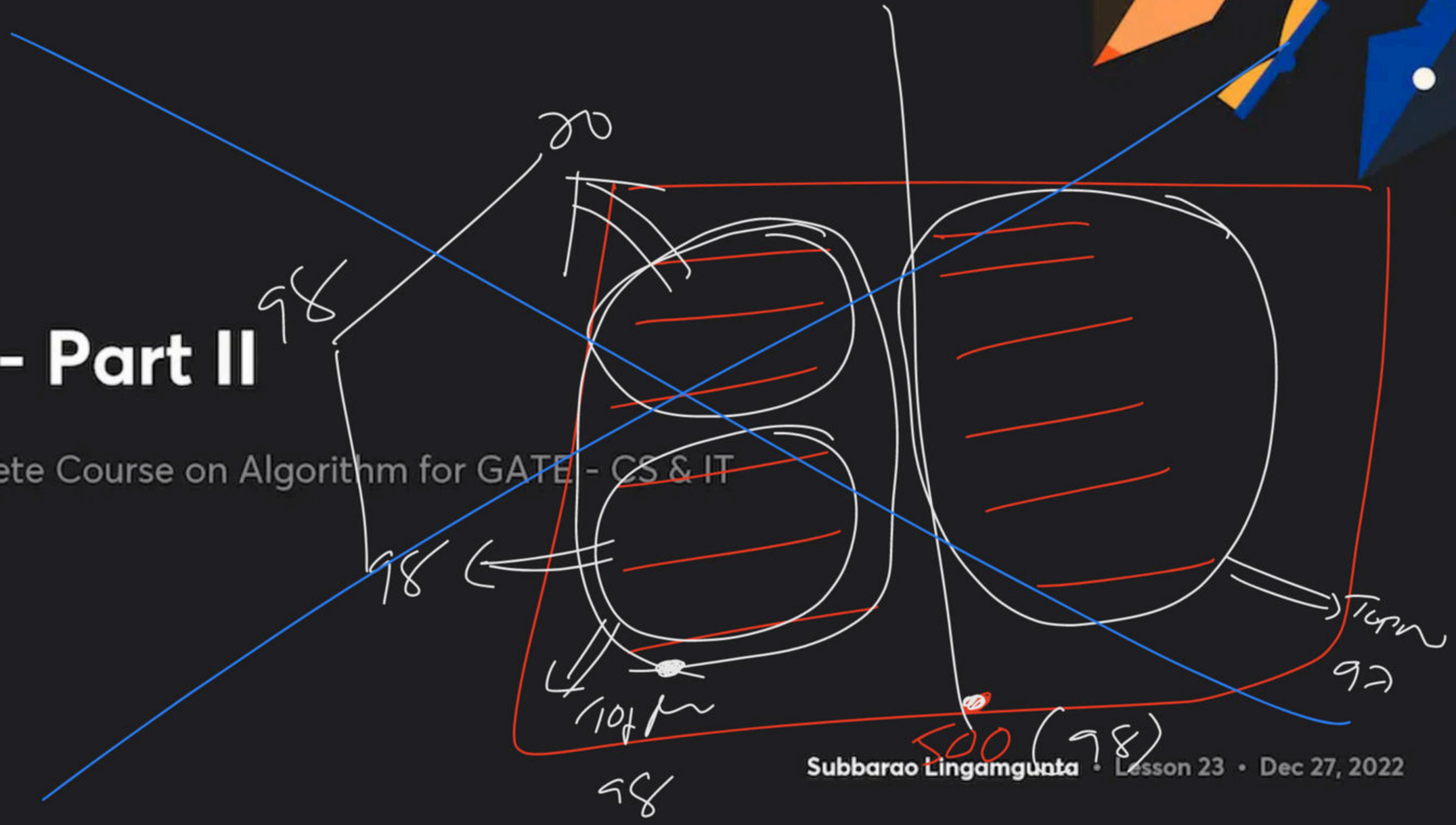
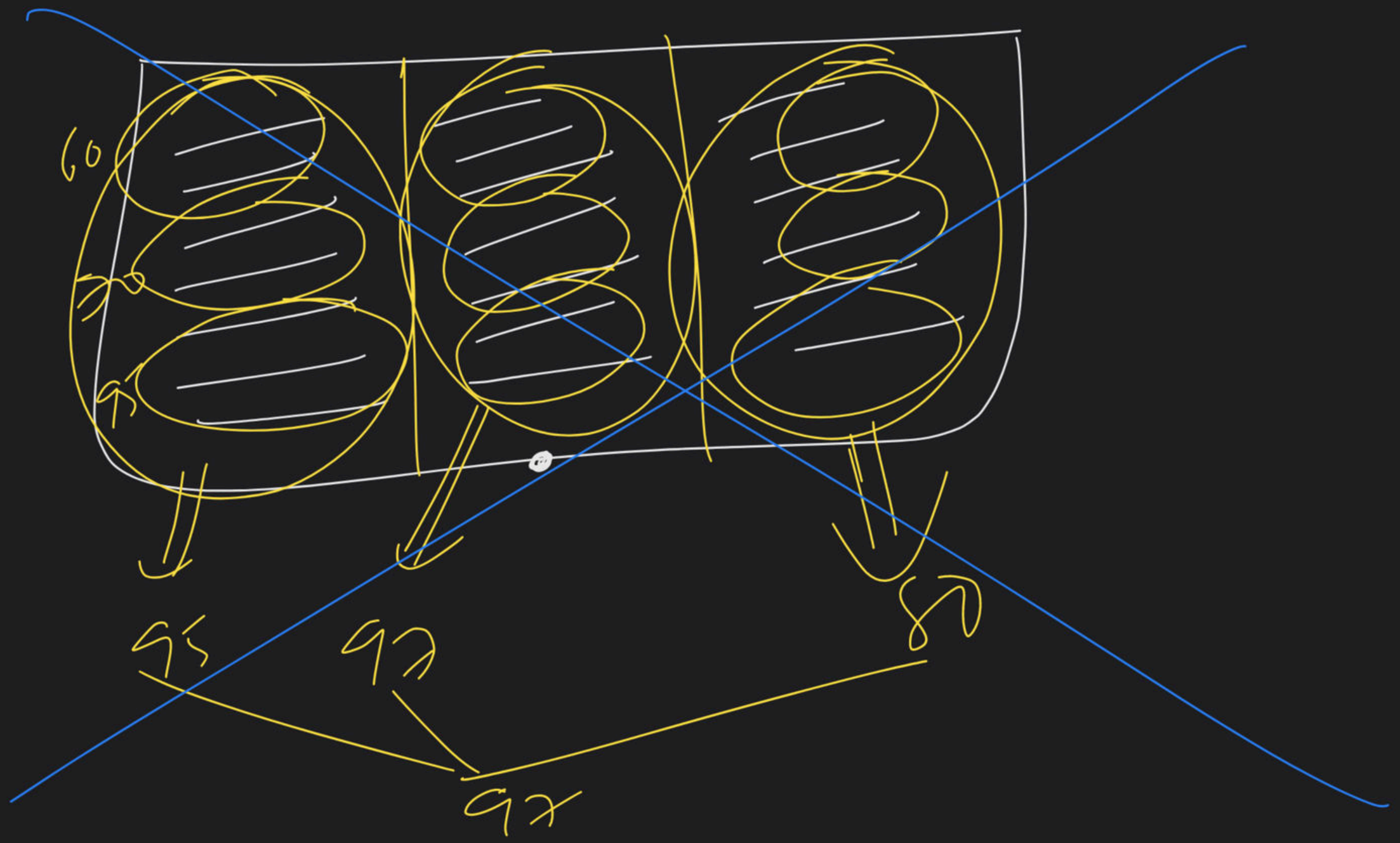
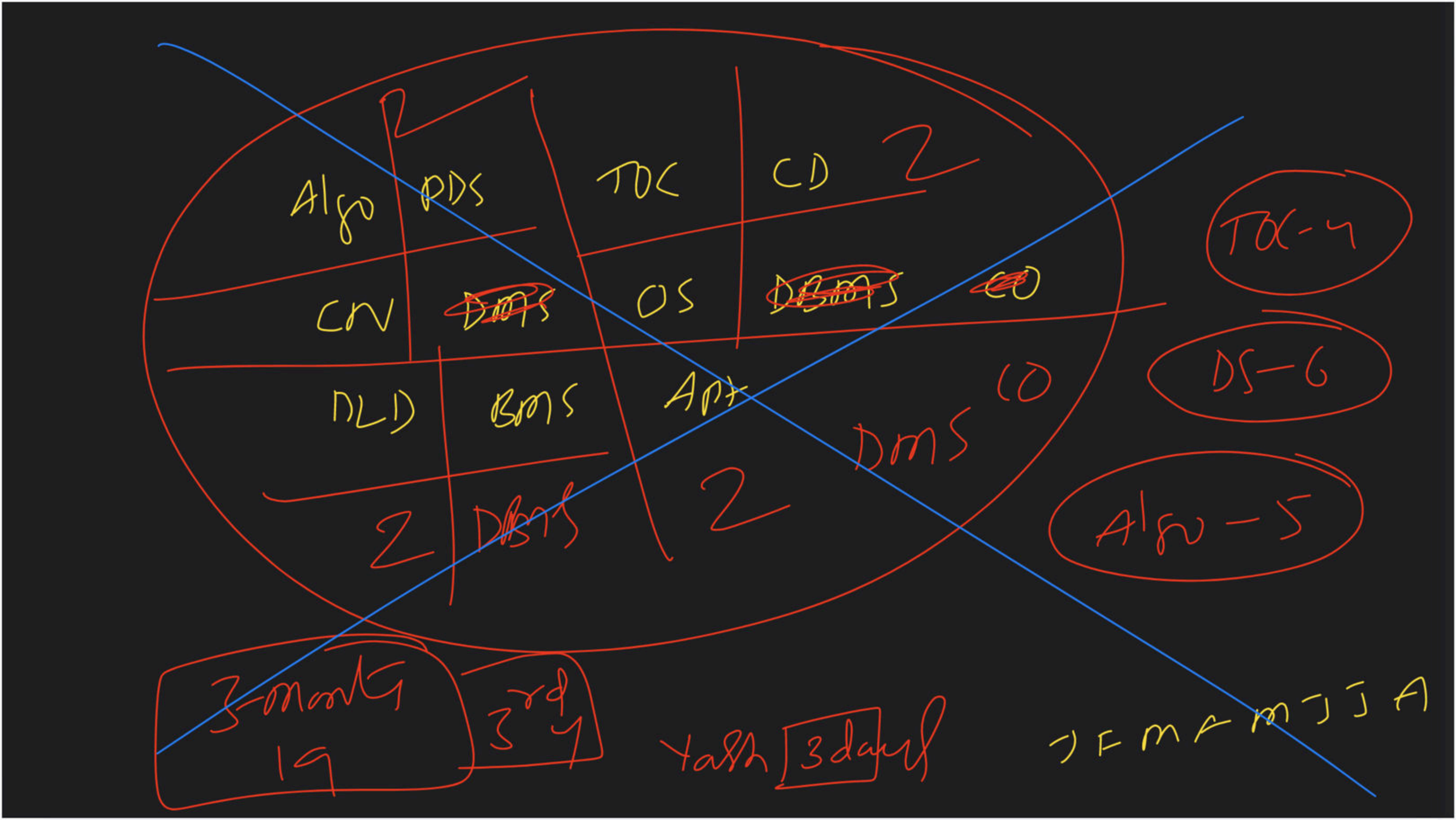
DP - Part II

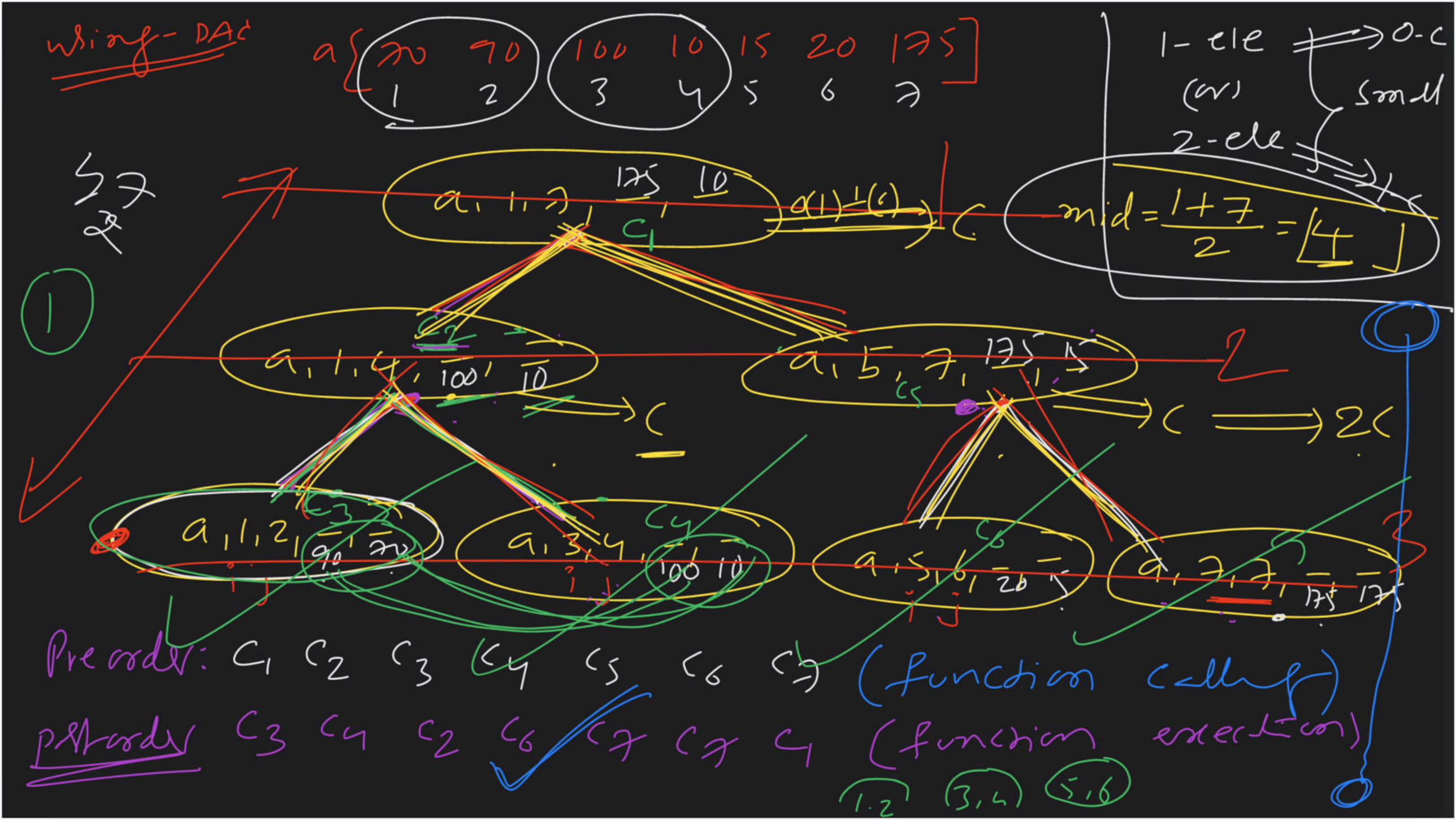
Complete Course on Algorithm for GATE - CS & IT



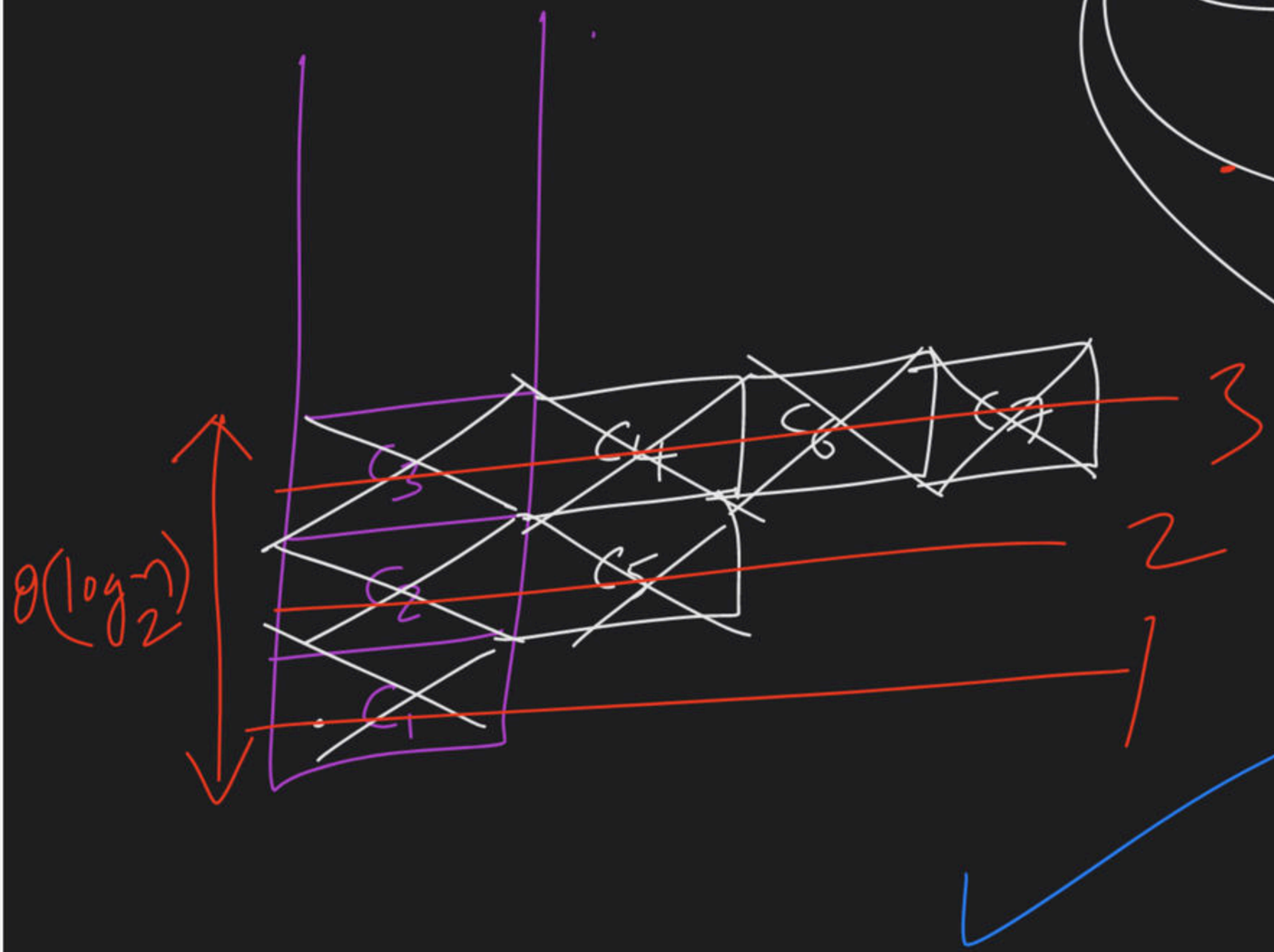
Subbarao Lingamgunta • Lesson 23 • Dec 27, 2022

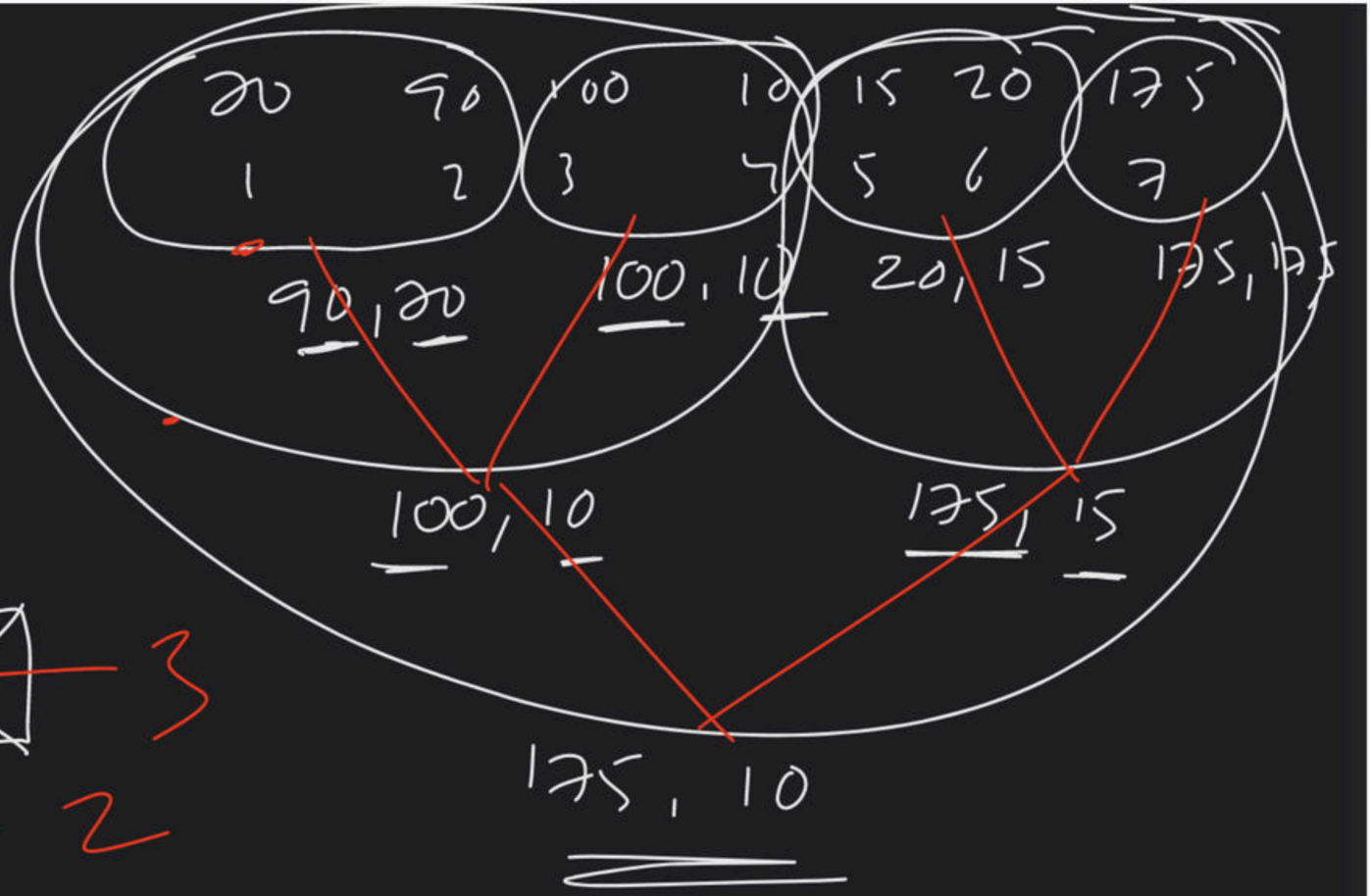


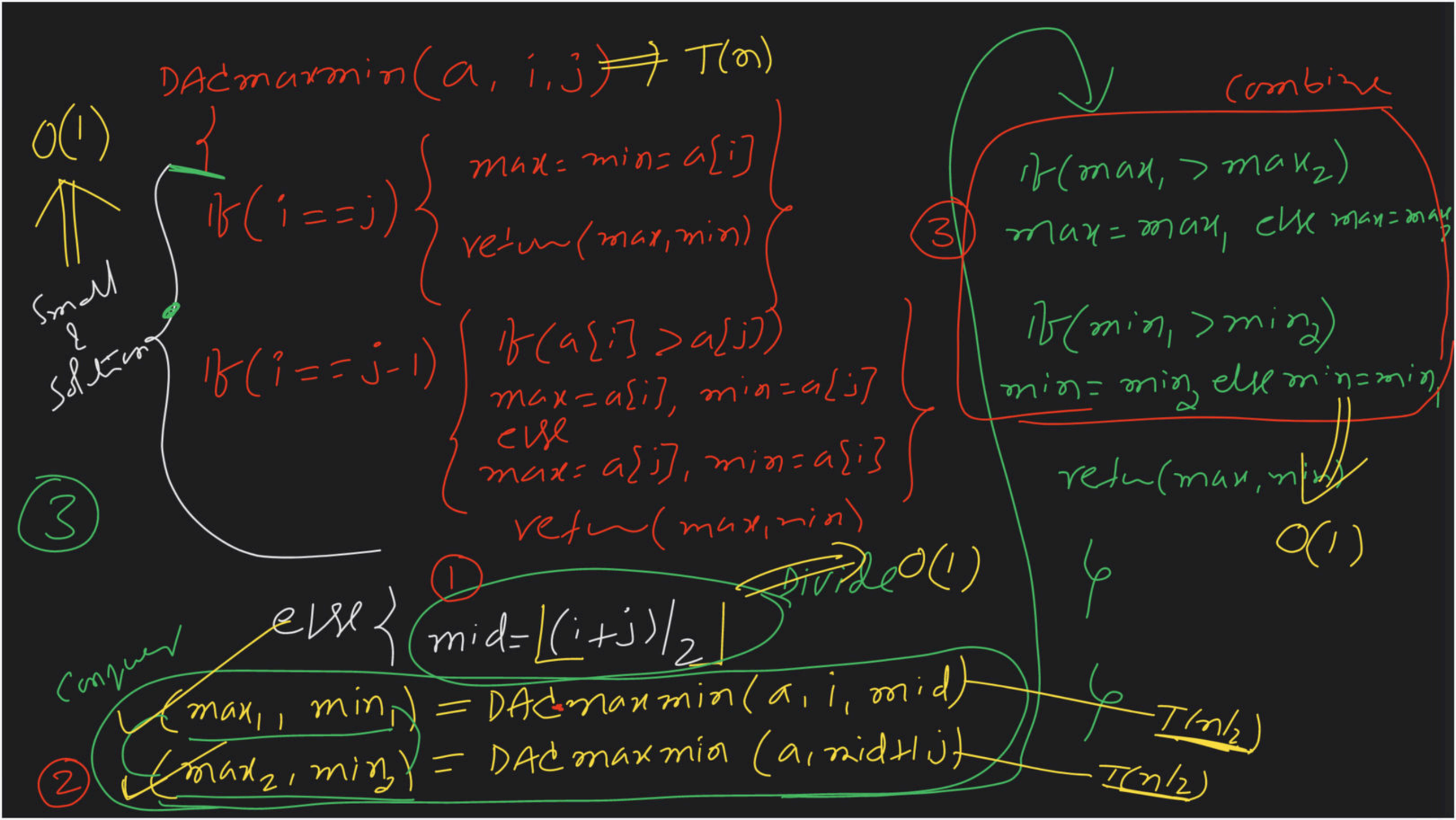

Algo
PDS
TOC
CD
CN
OS
NLD
BDMS
API
DMS
TOC-4
DS-6
Algo-5
3-months
19
3rd
Yash
3days
JFMAMJJA


Using - DAC
a[20 90 100 10 15 20 175]
1 2 3 4 5 6 7
1- ele (or) 2-ele
0-c
small
mid = 1+7/2 = 4
a, 1, 7, 175, 10
a, 1, 4, 100, 10
a, 5, 7, 175, 15
a, 1, 2, 90, 20
a, 3, 4, 100, 10
a, 5, 6, 20
a, 7, 7, 175, 175
Preorder: C1 C2 C3 C4 C5 C6 C7
(function calling)
postorder C3 C4 C2 C6 C7 C5 C1 (function execution)
1,2
3,4
5,6

2


C5
C4
C6
C7
C3
C2
C1
3
2
1
θ(log2)


20 90 100 10 15 20 175
1 2 3 4 5 6 7
90, 20
100, 10
20, 15
175, 15
100, 10
175, 15
175, 10


DACmaxmin(a, i, j) ⟹ T(n)
O(1)
Small & Solution
if(i==j) { max=min=a[i] return(max,min) }
if(i==j-1) { if(a[i]>a[j]) max=a[i], min=a[j] else max=a[j], min=a[i] return(max,min) }
3
else {
1 mid=⌊(i+j)/2⌋ Divide O(1)
2 (max1, min1) = DACmaxmin(a, i, mid)
(max2, min2) = DACmaxmin(a, mid+1, j)
Conquer
T(n/2)
T(n/2)
Combine
3 if(max1 > max2) max=max1 else max=max2
if(min1 > min2) min=min2 else min=min1
return(max,min)
O(1)

TC of above program

RR - Time

$$T(n) = \begin{cases} O(1) & \text{if } n = 1 \text{ (or) } n = 2 \\ O(1) + 2T(n/2) + O(1) & \text{if } n > 2 \end{cases}$$

$$T(n) = 2T(n/2) + C$$

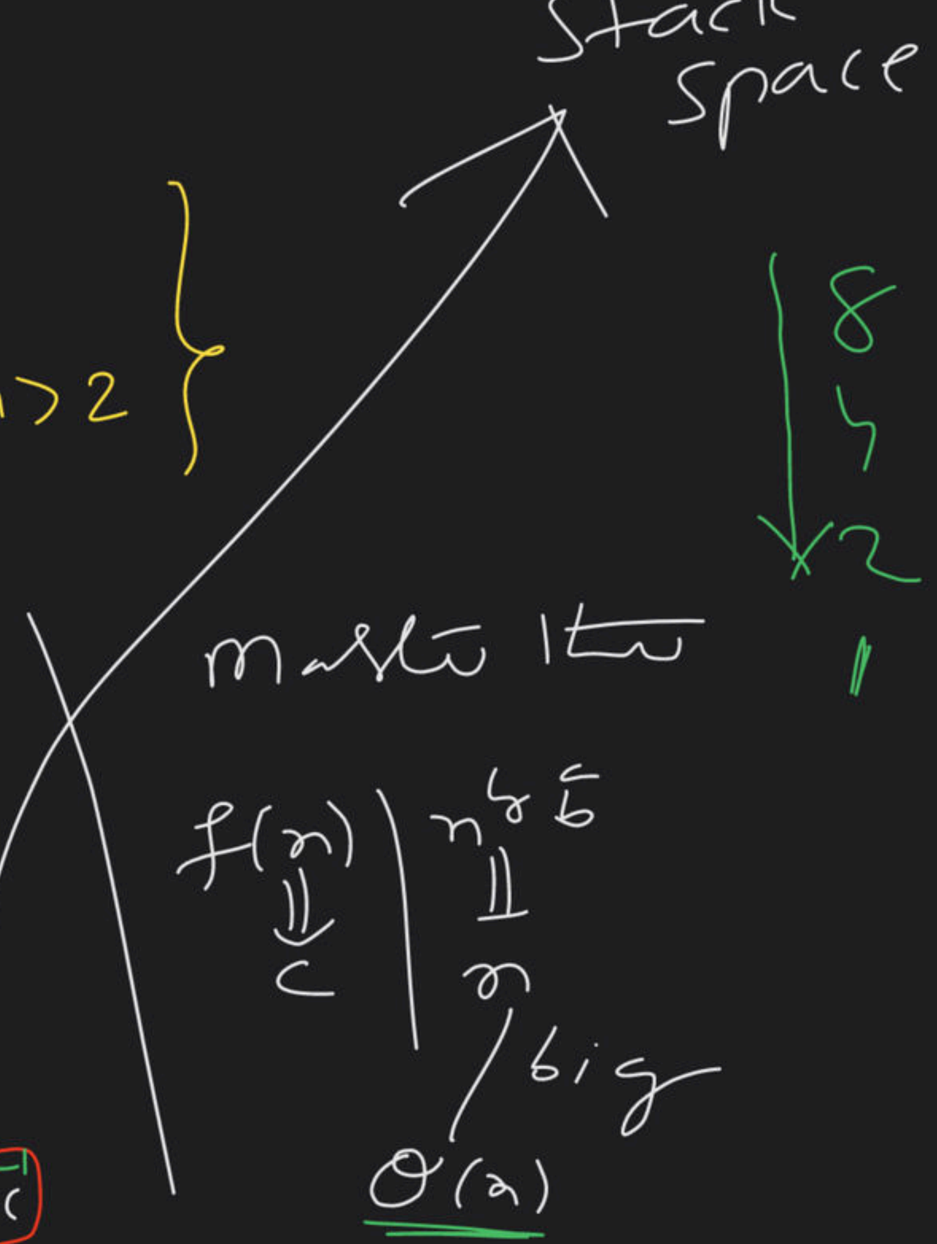
$$= 2\{2T(n/2^2) + C\} + C$$

$$= 2^2 + (n/2^2) + 2C + C$$

$$= 2^3 + (n/2^3) + 2^2C + 2^1C + 2^0C$$

$$= 2^{\log n} T(1) + \underbrace{2^0C + 2^1C + \cdots + 2^{\log n - 1}C}_{\log n}$$

(4)



$= 2^{\frac{n}{2}-1} T(1) + c[2^0 + 2^1 + \cdots + 2^{\frac{n}{2}-1}]$

$= \frac{n}{2} \cdot T(2) + c \cdot 2^{\frac{n}{2}}$

$= \frac{n}{2} \cdot O(1) + c \cdot n$

Leaf node cost

$= \frac{n}{2} \cdot c + c \cdot n \Longrightarrow \Theta(n)$

$\frac{2^0(2^{\frac{n}{2}}-1)}{2-1}$

$2^{\frac{n}{2}}$

(5)

Space complexity

=

Stack space

=

$\Theta(\log n)$.