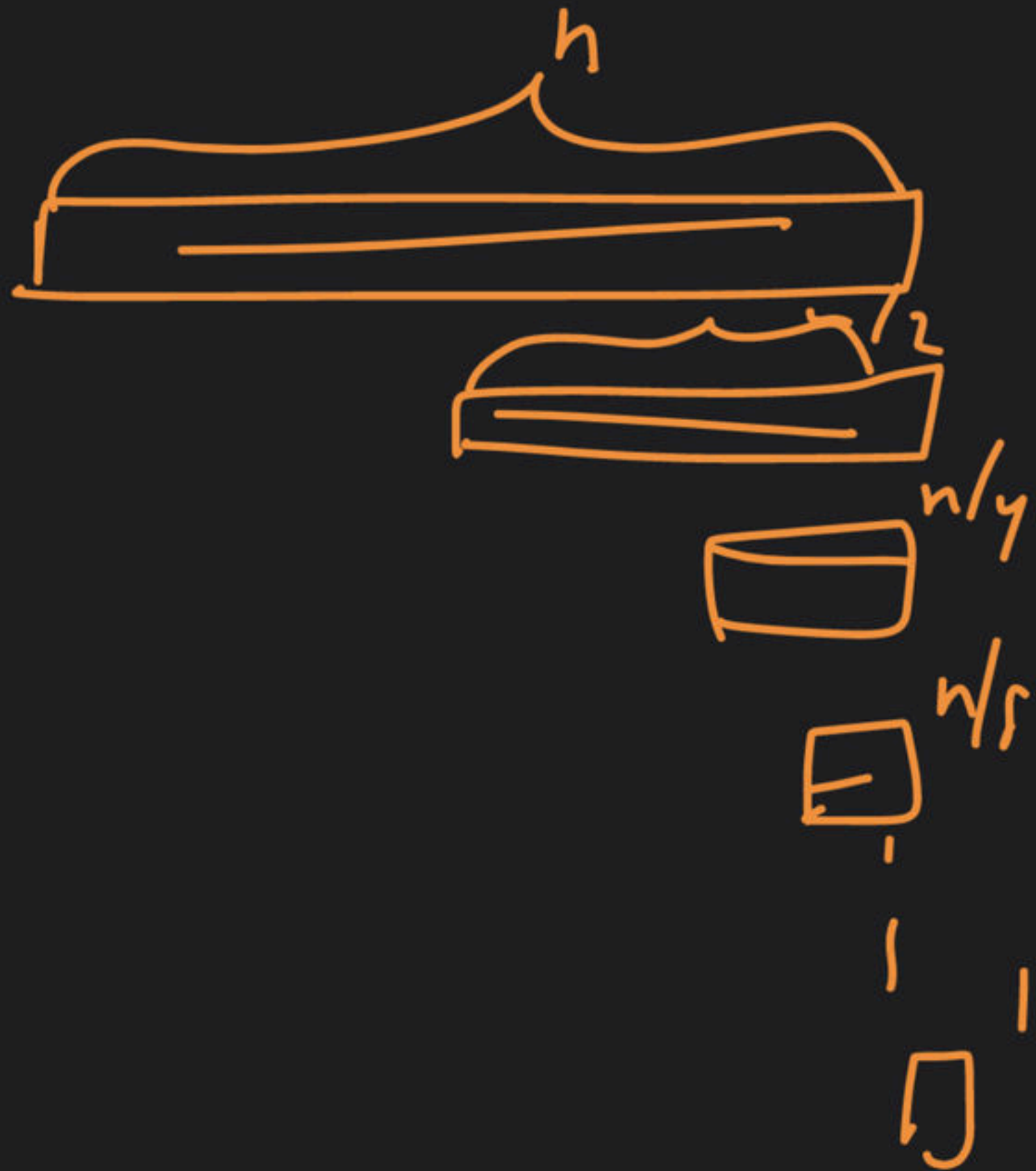


DnC Class - 1

Special class

→ Merge Sort



DnC

Angrez 2

Single element array
is considered as SORTED

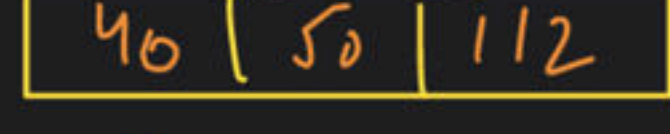
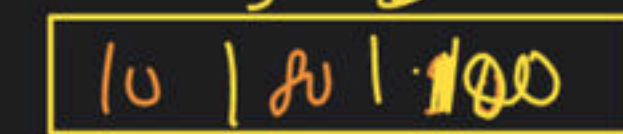
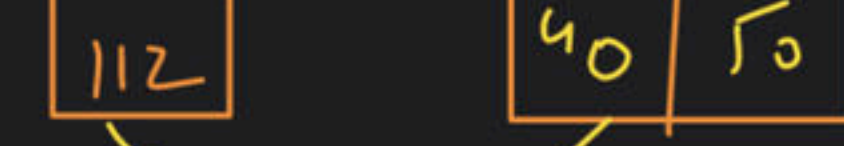
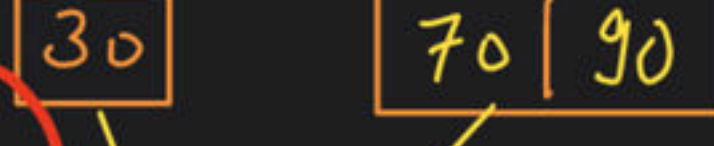
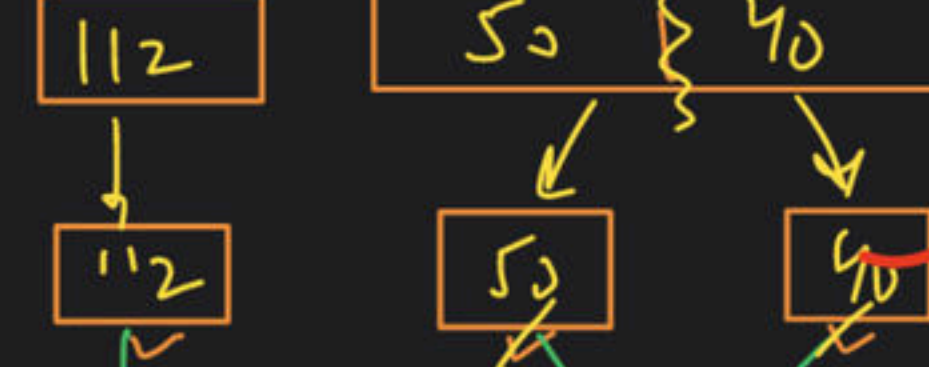
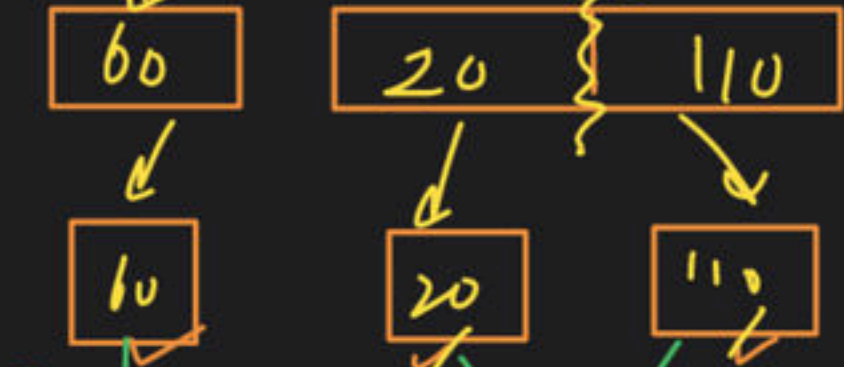
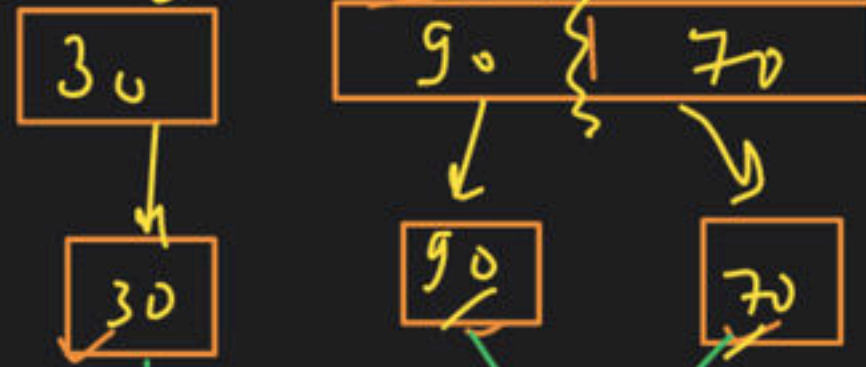
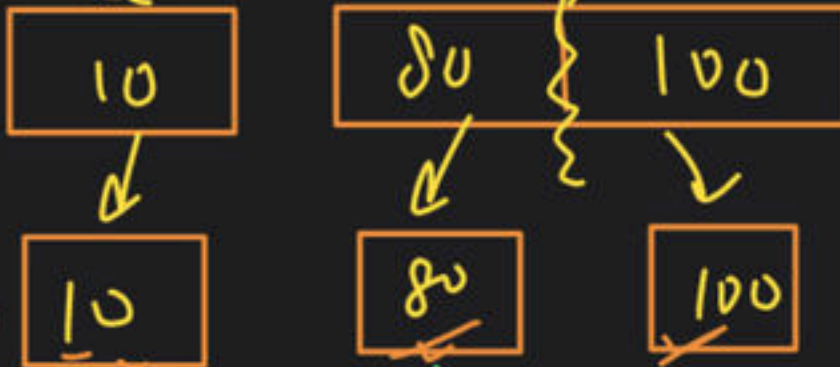
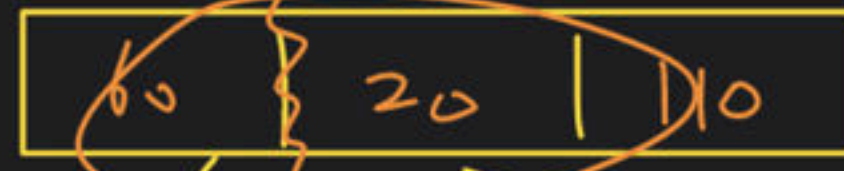
✓ 30%.
X

Top 70%
X

1 week

5-10%/-
top

last week \Rightarrow Top 5-10%/-



2 min
3 calls

merge
 $O(n \log n)$

initial Array

left

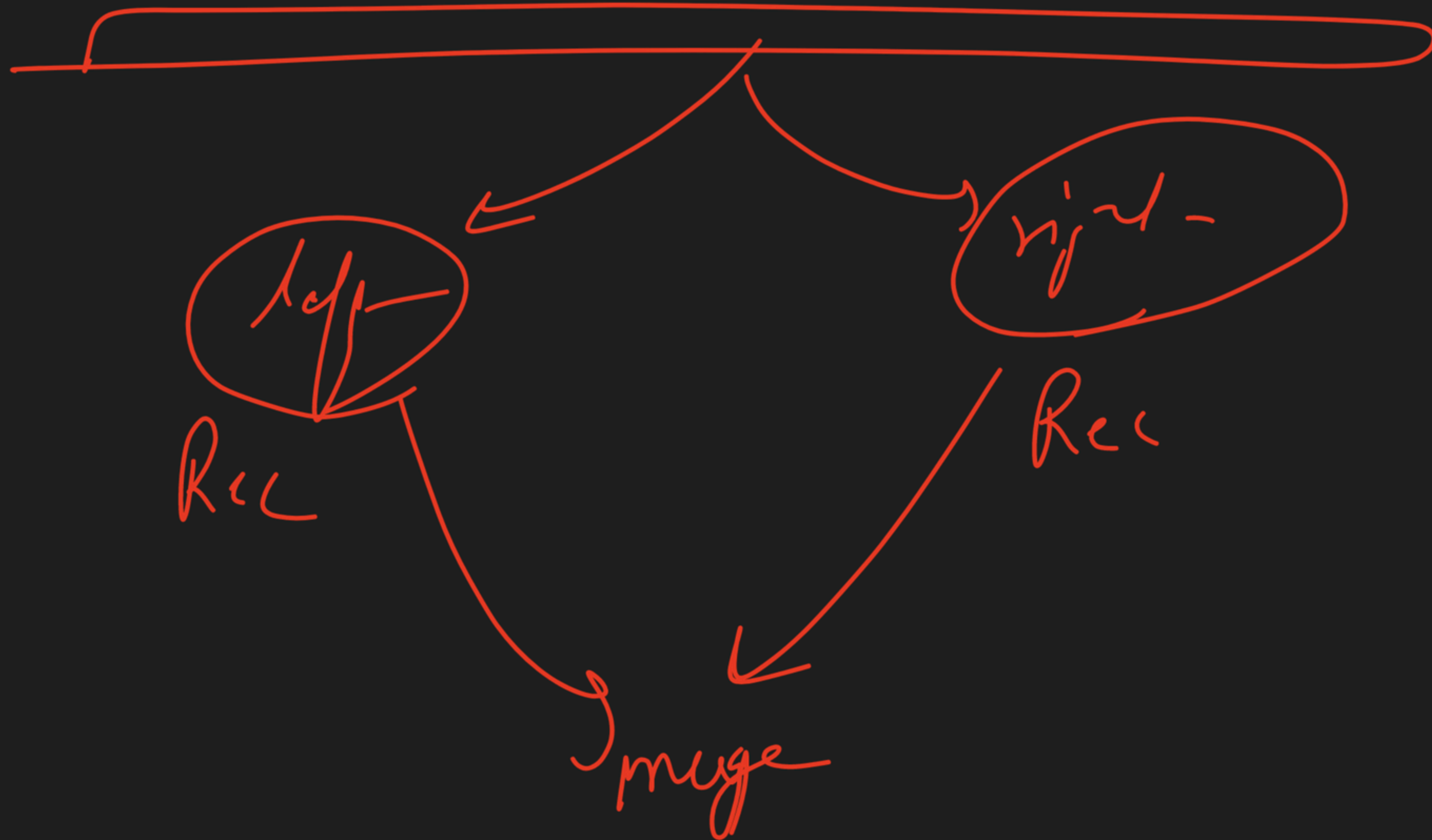
Rec

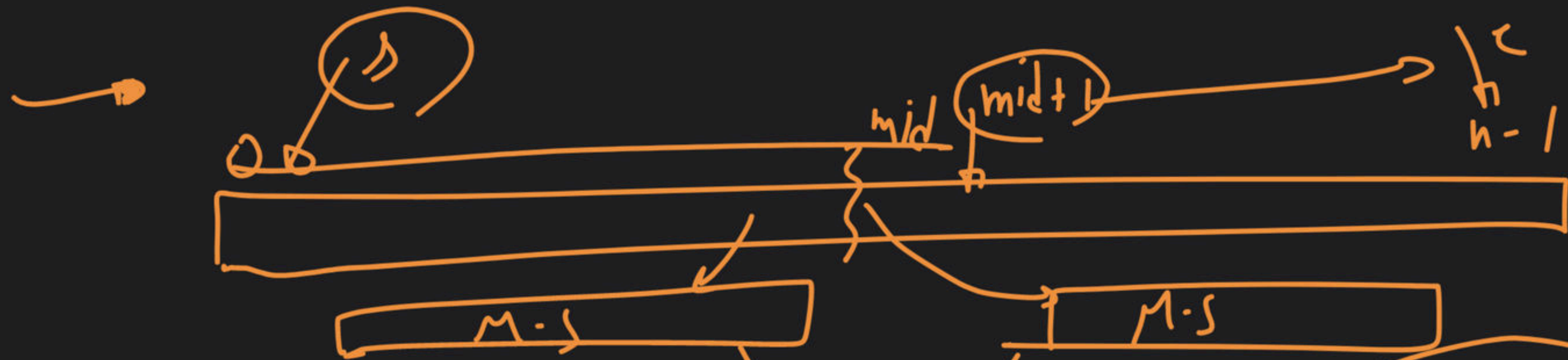
sort

right

Rec

2 sorted
Merge





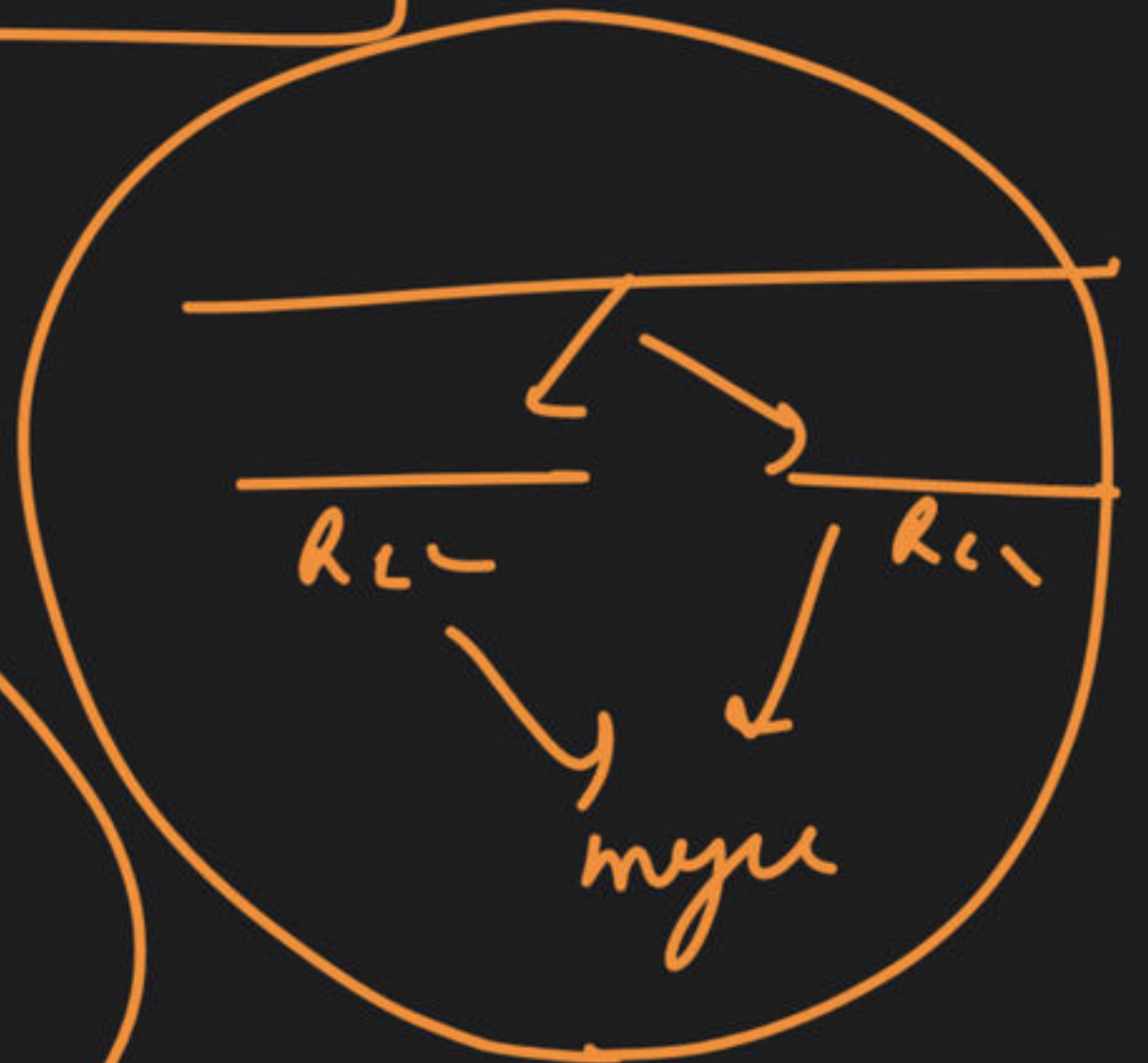
\rightarrow mergeSort(arr, s, e)

merge

mergeSort(arr, s, mid);

mergeSort($arr, mid+1, e$)

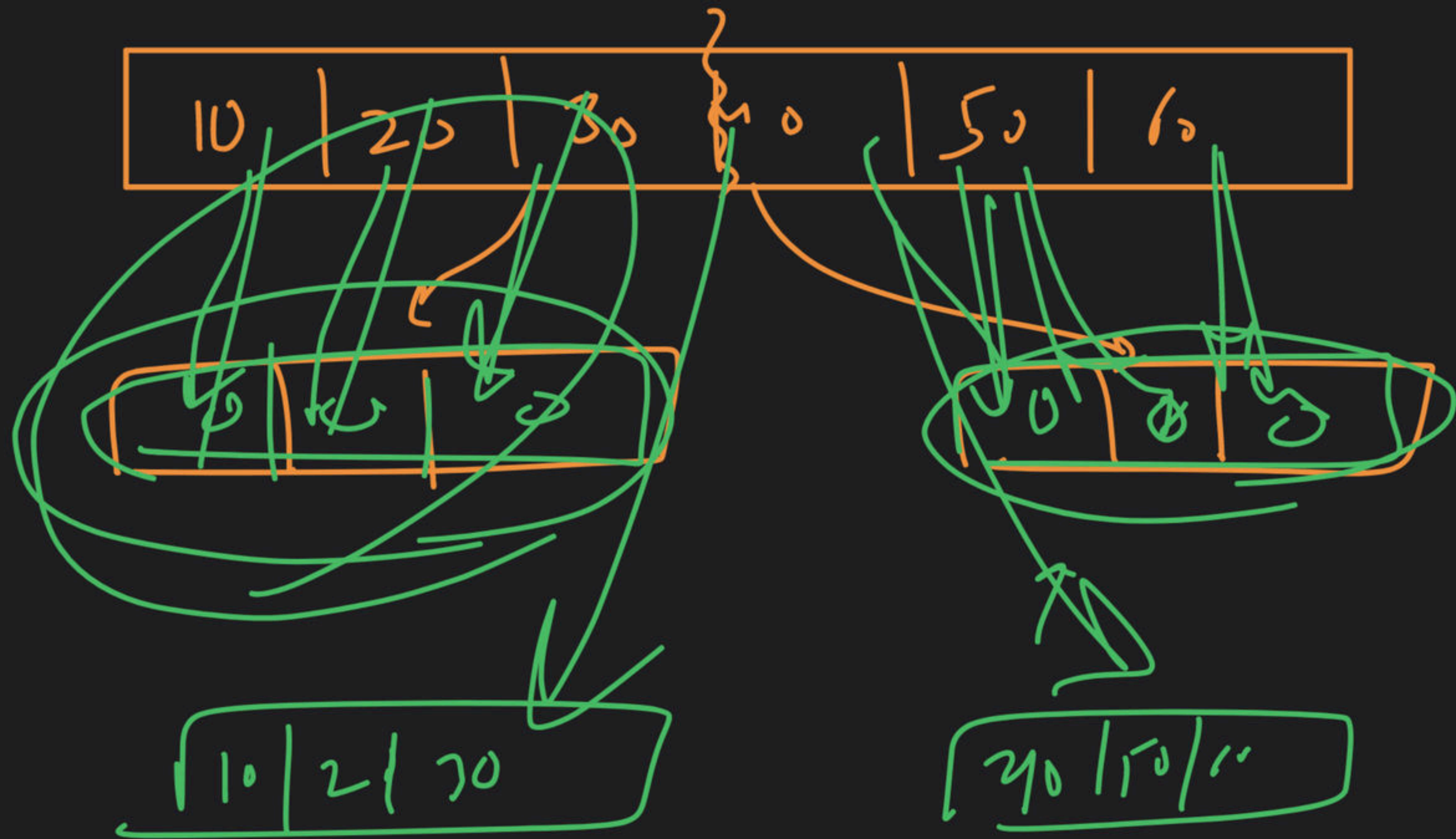
merge(arr, s, e, mid)



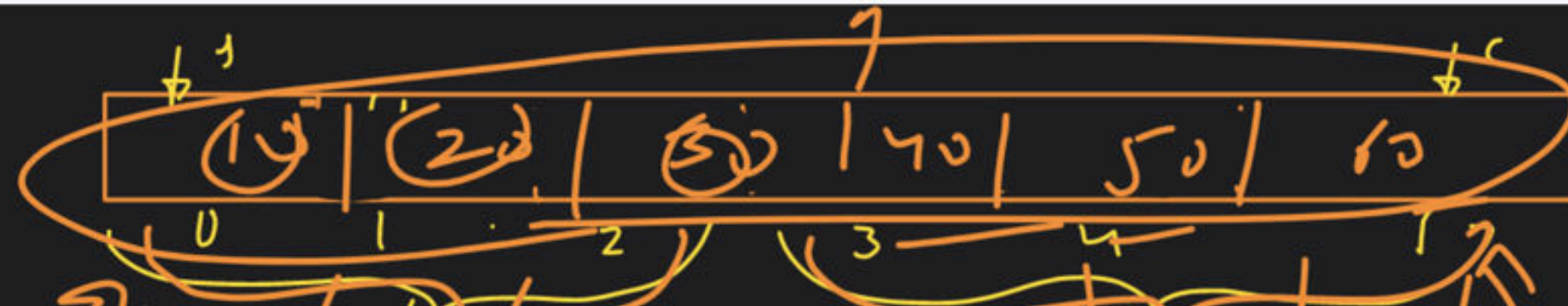


$$\text{leftLen} = 4 - 0 + 1 \rightarrow \text{mid} - \text{start} + 1$$

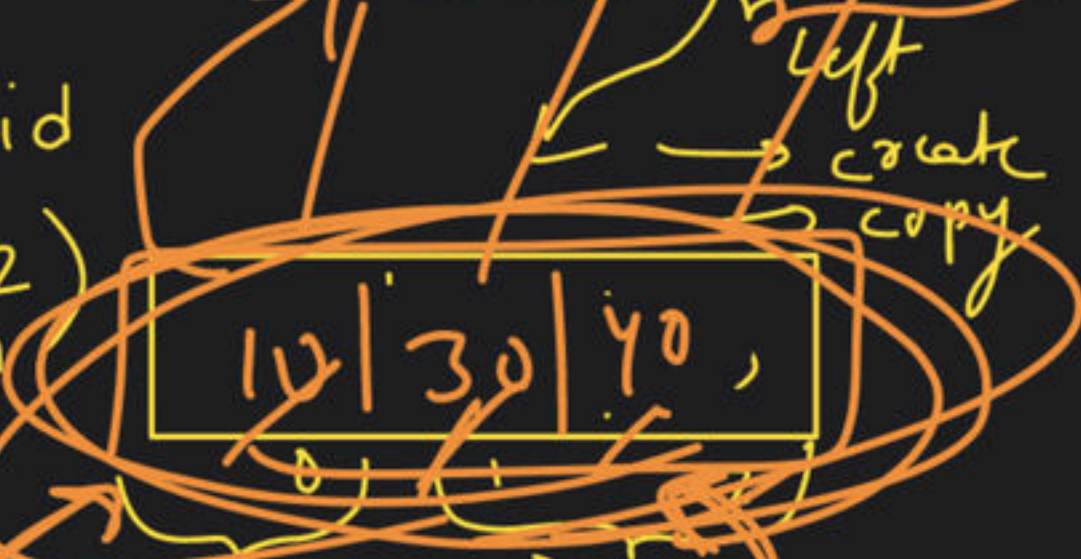
$$\text{rightLen} = 8 - 4 \rightarrow \text{end} - \text{mid}$$



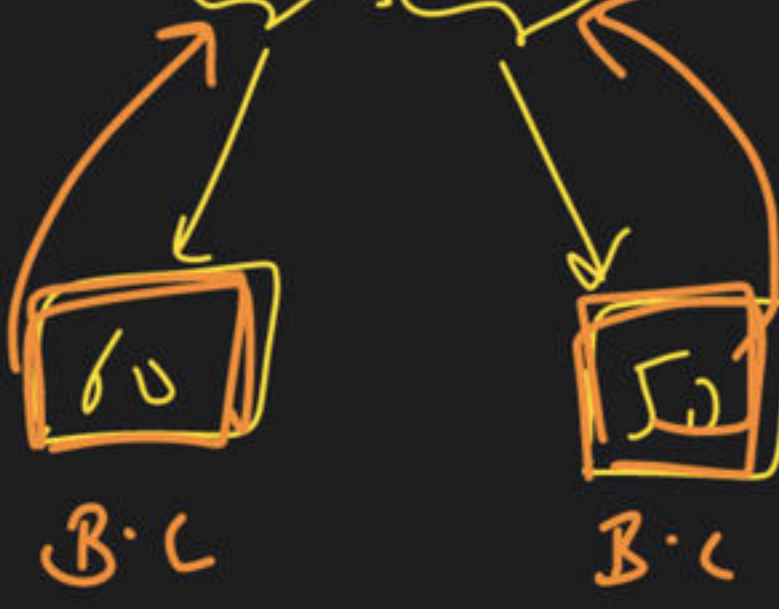
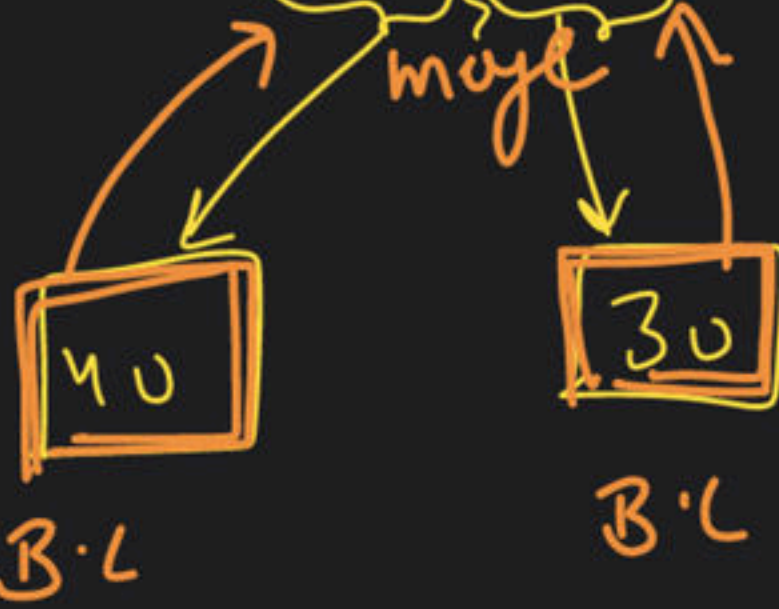
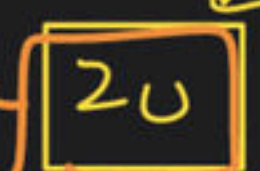
M.S (arr, 0, 5)



M.S (arr, 0, 2)



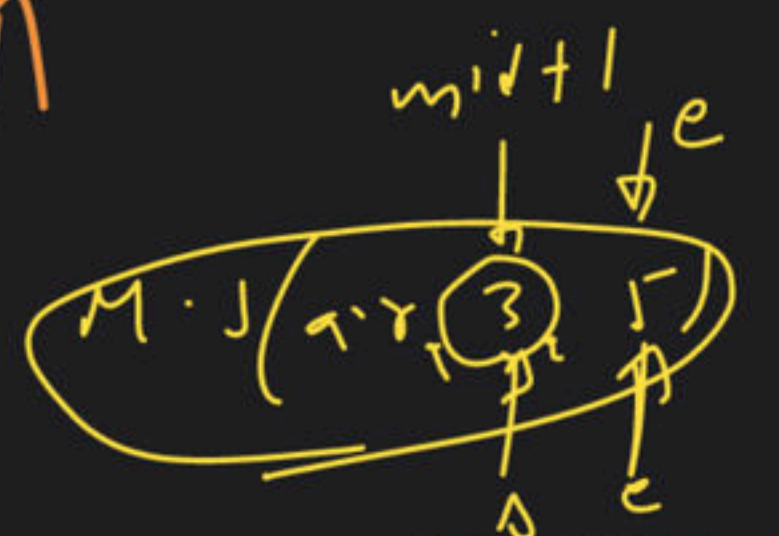
$$2 - 0 + 1 = 3$$



Rec

Rec

mid = 2



May

divide



$$a = \log_4$$

\leftarrow

$$\begin{aligned}
 T(n) &= \cancel{a} + \cancel{2T\left(\frac{n}{2}\right)} + \cancel{\left(\frac{nK}{2}\right)} \\
 2T\left(\frac{n}{2}\right) &= \cancel{2} + \cancel{4T\left(\frac{n}{4}\right)} + \cancel{\left(\frac{nK}{2}\right)} \\
 4T\left(\frac{n}{4}\right) &= \cancel{4} + \cancel{8T\left(\frac{n}{8}\right)} + \cancel{\left(\frac{nK}{2}\right)} \\
 8T\left(\frac{n}{8}\right) &= \cancel{8} + \cancel{16T\left(\frac{n}{16}\right)} + \cancel{\left(\frac{nK}{2}\right)} \\
 &\vdots \\
 T(1) &= \cancel{a} \\
 \hline
 T(n) &= a \times nK
 \end{aligned}$$

$\begin{matrix} \text{B.C.} \\ \text{1st Rec} \\ \text{2nd Rec} \\ \text{3rd Rec} \end{matrix}$

$\left. \begin{aligned} &+ \left(\frac{nK}{2}\right) \\ &+ \left(\frac{nK}{2}\right) \\ &+ \left(\frac{nK}{2}\right) \\ &+ \left(\frac{nK}{2}\right) \\ &\vdots \\ &+ (nK) \end{aligned} \right\} a-1$

$$T(n) = \textcircled{a} \times n \times X$$

$$= \log n \times n$$

$$T(n) = n * \log n$$

Code

Class \rightarrow Complexity

left / right \rightarrow create / copy
 α / α

\rightarrow 1/w

Inplace merge Sort

1/w





Merge 2 sorted Arrays

i/p

arr

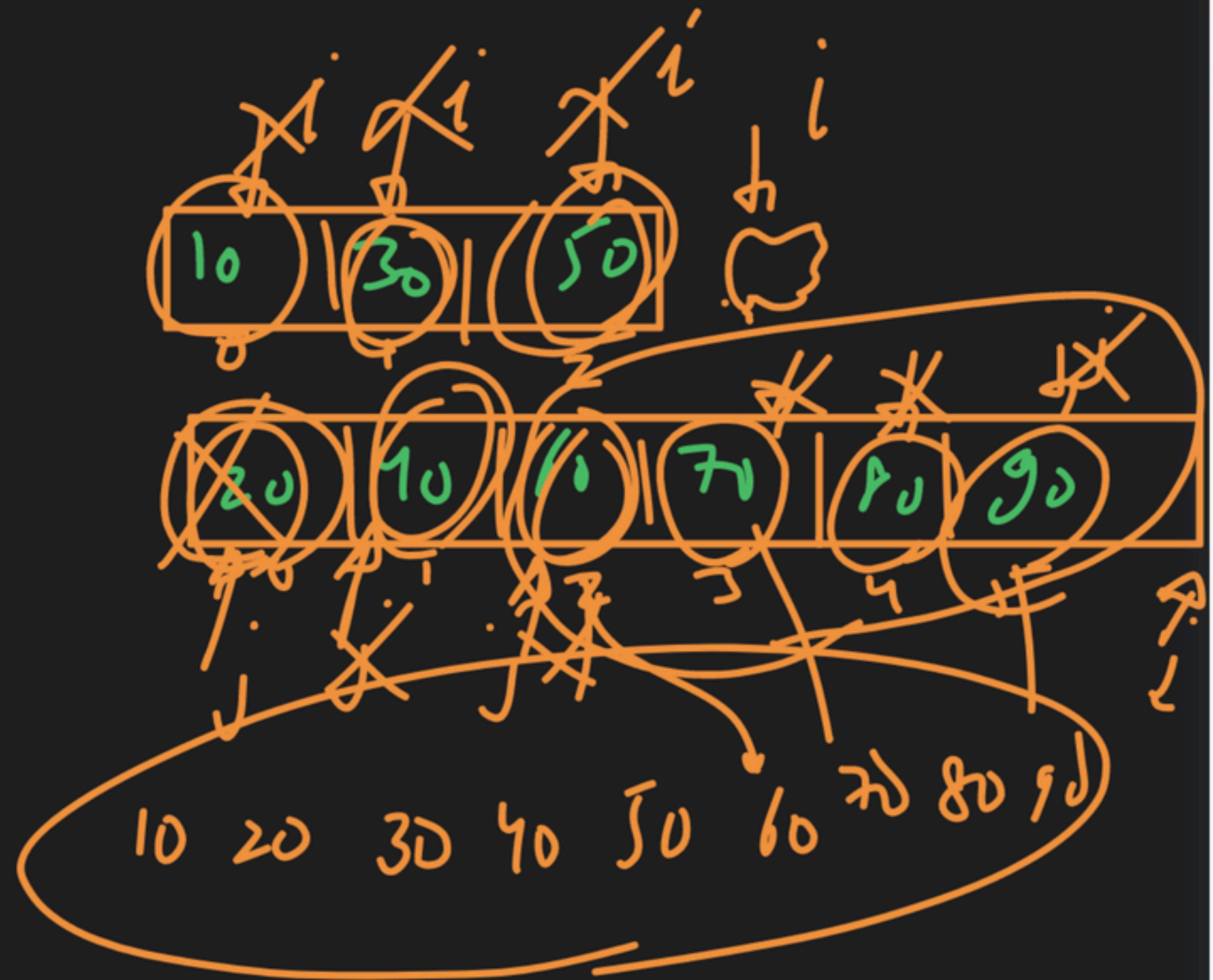
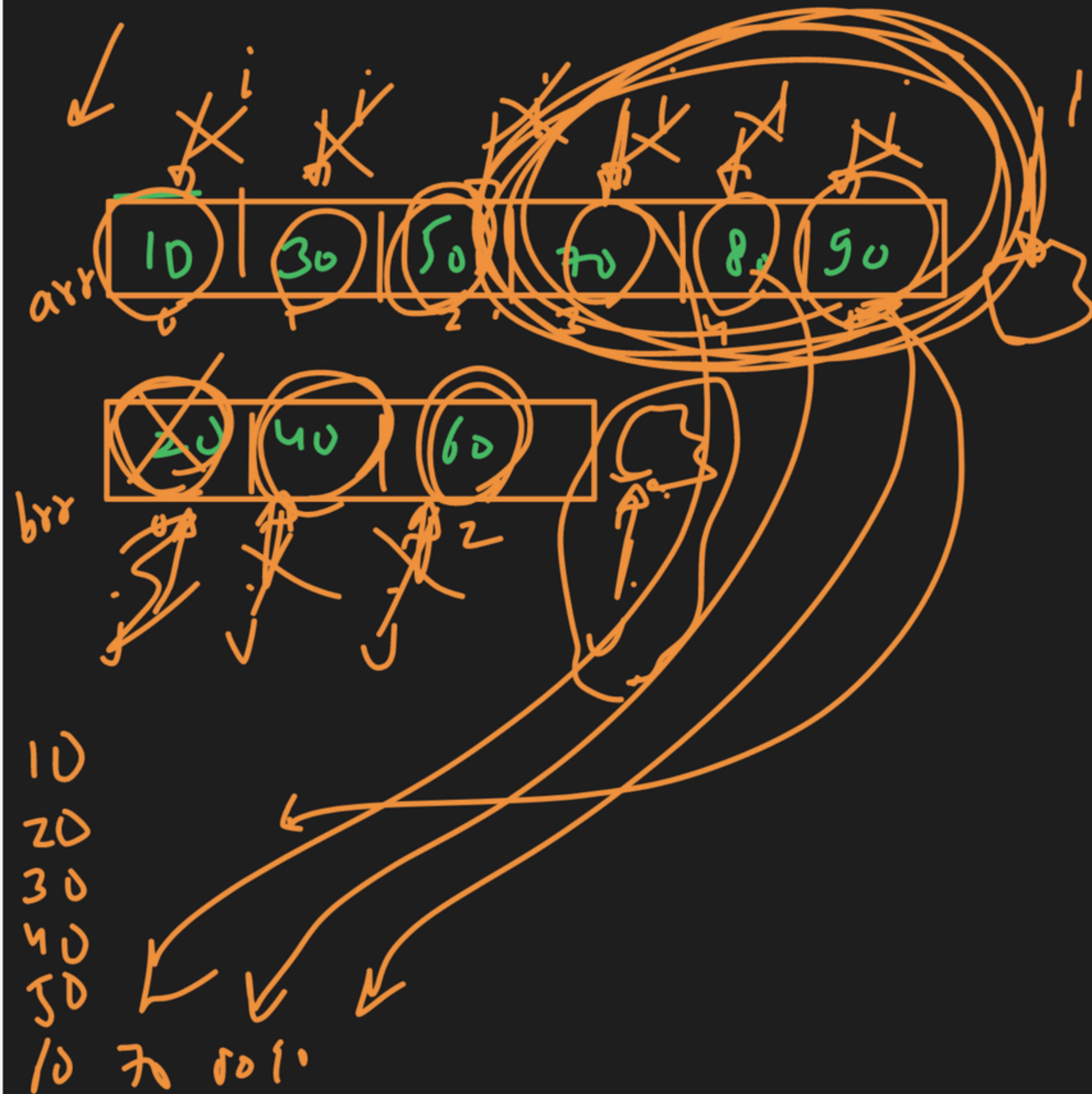
20	40	60	80	90	100
----	----	----	----	----	-----

brr

10	30	50	70		
----	----	----	----	--	--

10	20	30	40	50	60	70	80	90	100
----	----	----	----	----	----	----	----	----	-----

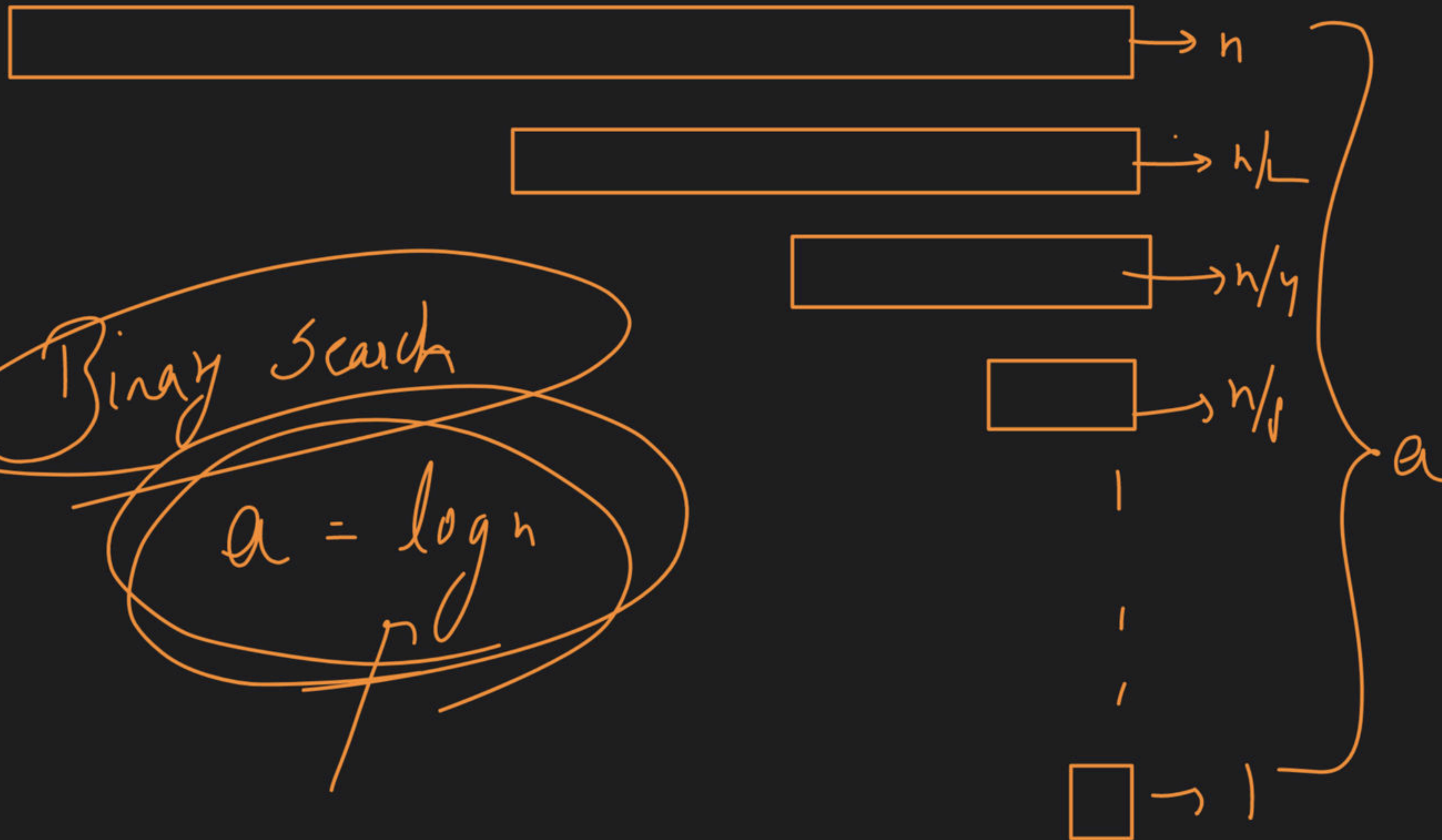
ans

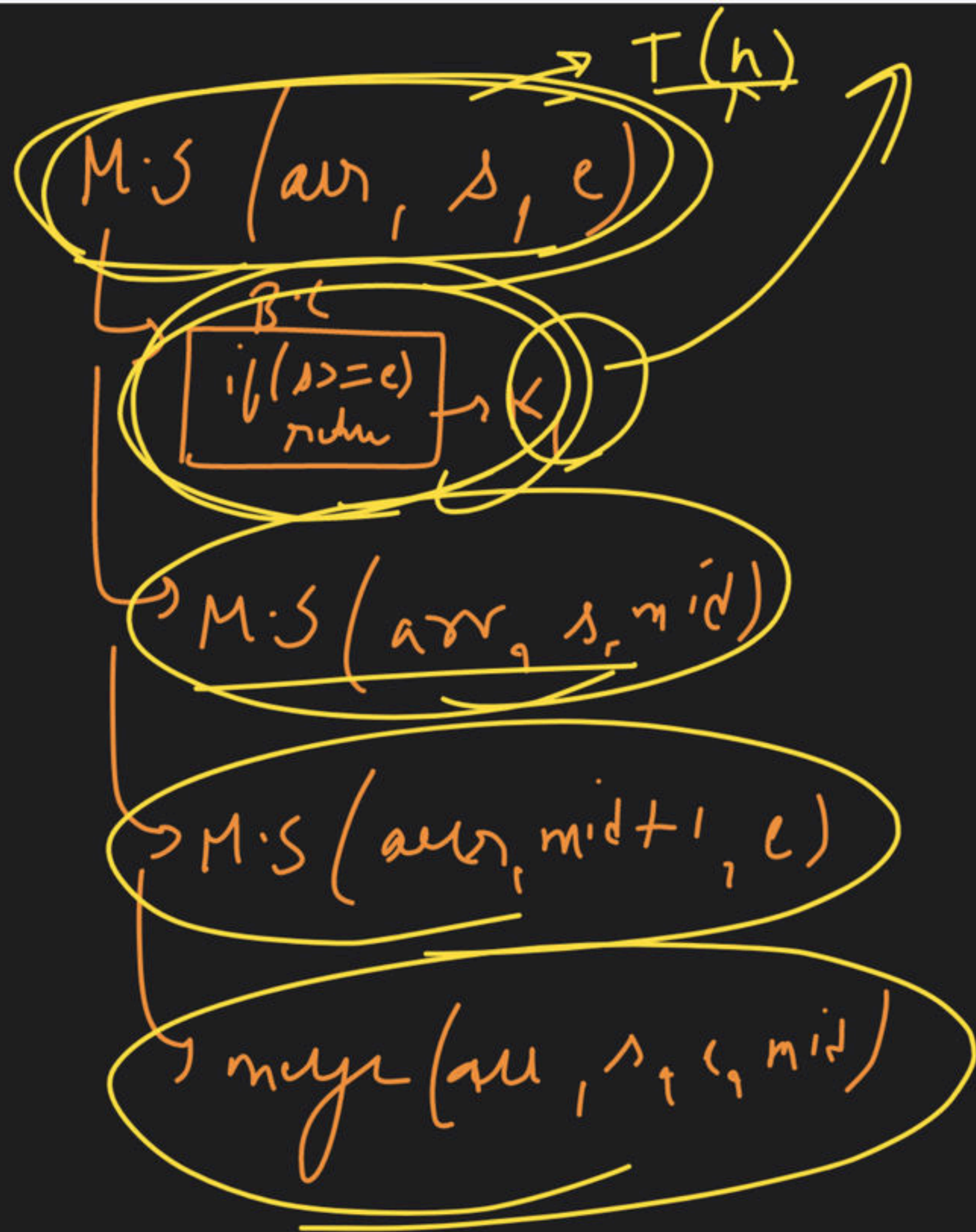


Binary Search

$a = \log n$

n^0





$$T(n) = K_1 + T\left(\frac{n}{2}\right) + T\left(\frac{n}{2}\right) + (n \star K)$$

$$2T\left(\frac{n}{2}\right)$$

$$a + a$$

$$= 2a$$

$$K + K$$

$$= 2K$$

$$T(n) = k_1 + 2T\left(\frac{n}{2}\right) + (n \star k)$$

$$2 \times \left[2T\left(\frac{n}{2}\right) = 2k_1 + 1T\left(\frac{n}{4}\right) + 2 \times \left(\frac{n \star k}{2}\right) \right]$$

$$4 \times \left[4T\left(\frac{n}{4}\right) = 4k_1 + 8T\left(\frac{n}{8}\right) + 4 \times \left(\frac{n \star k}{4}\right) \right]$$

$$8 \times \left[8T\left(\frac{n}{8}\right) = 8k_1 + 16T\left(\frac{n}{16}\right) + 8 \times \left(\frac{n \star k}{8}\right) \right]$$

$$T(1) = k_1$$

$\rightarrow (n-1)$

a time

$$T(n) = (a-1) \star (n \star k)$$

$$T(n) = (a - 1) \star (n) \star \frac{1}{\star}$$

g.s./.

$$= a \star n$$

$$= (\log n) \times (n)$$

$$T(n) = n \star \log n$$

Space
Complexity

Extra Class



Kah

8:30 pm

















































