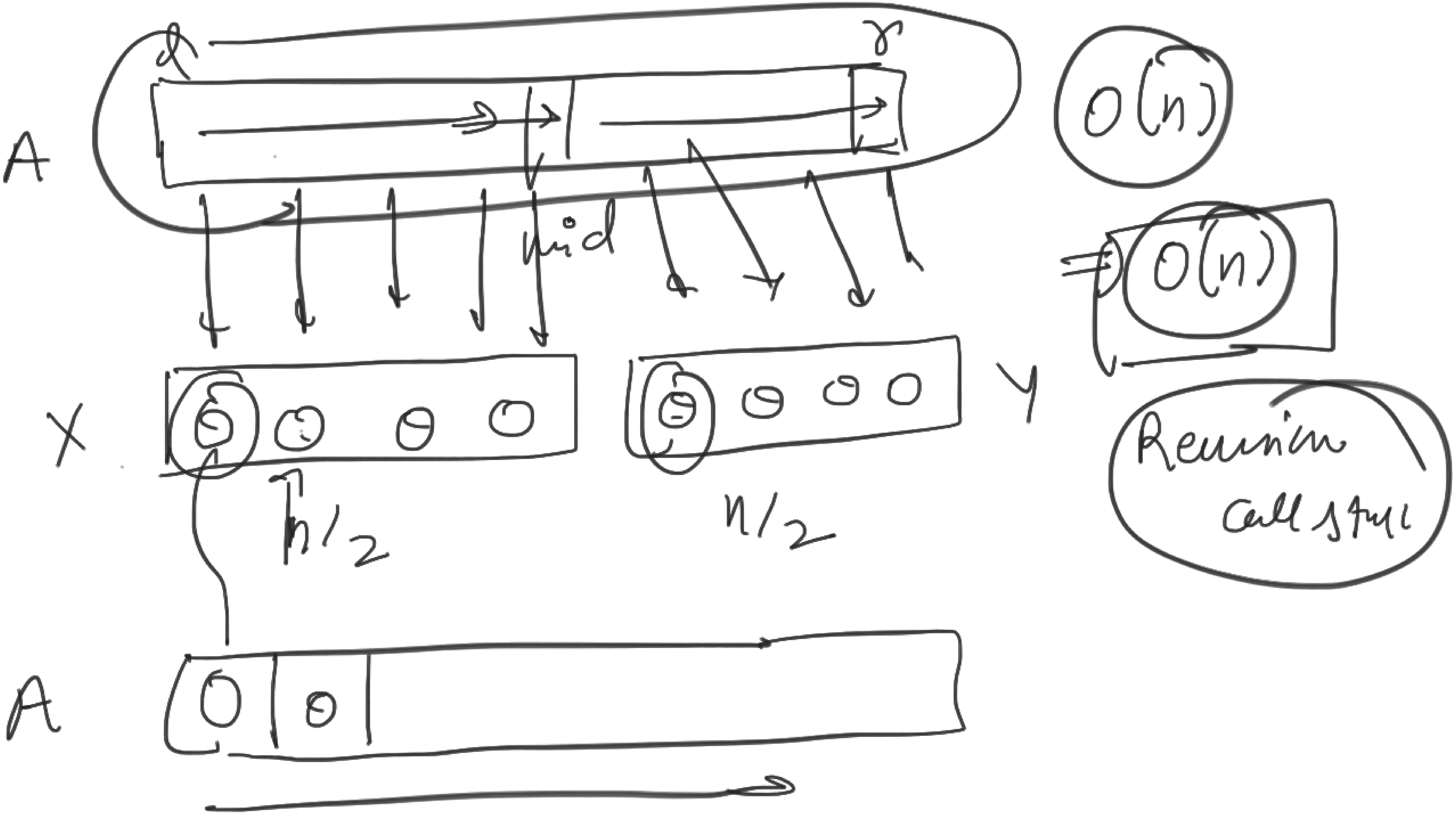
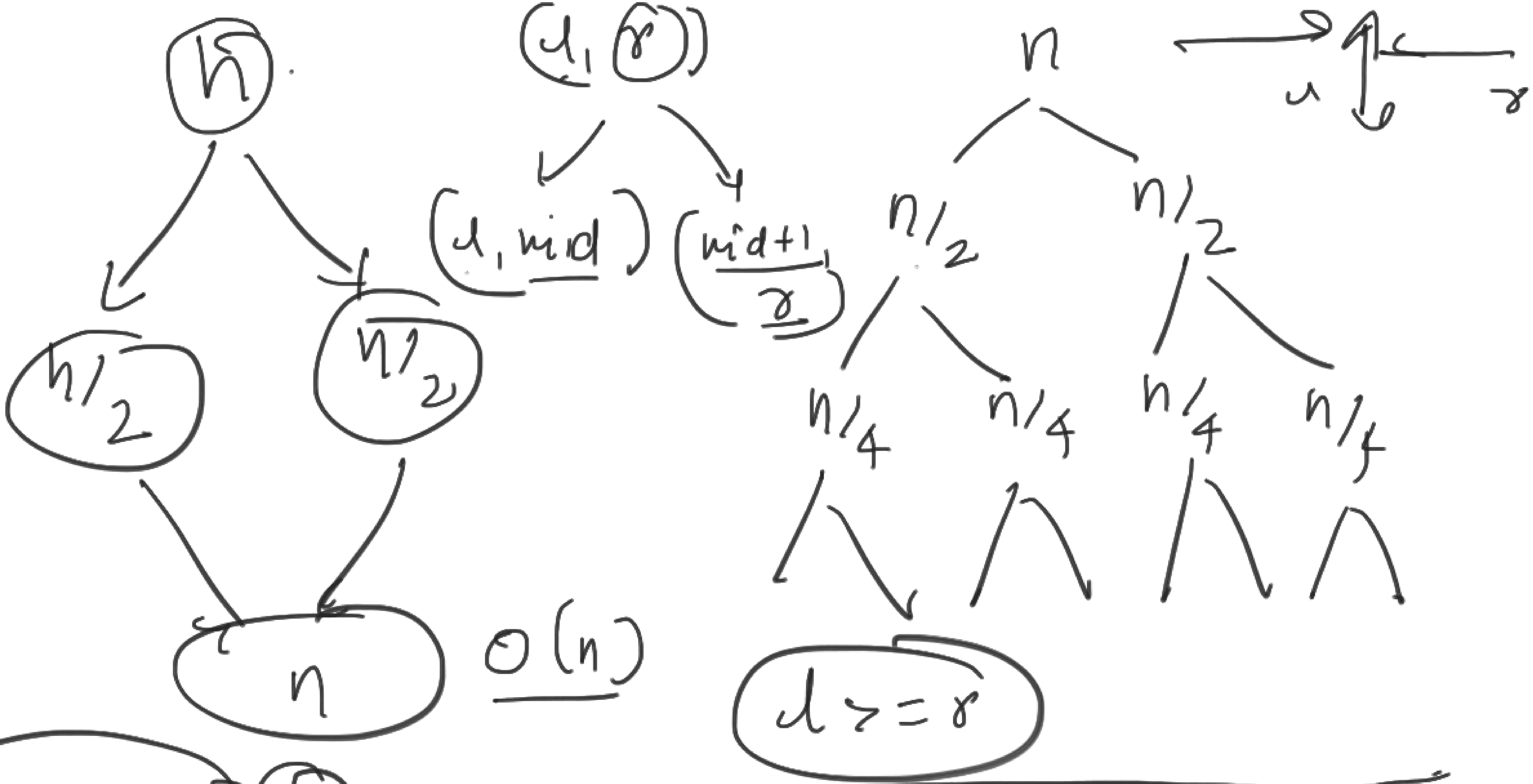

A
l
r
mid
O(n)
O(n)
X
Y
n/2
n/2
Recursion call stack
A



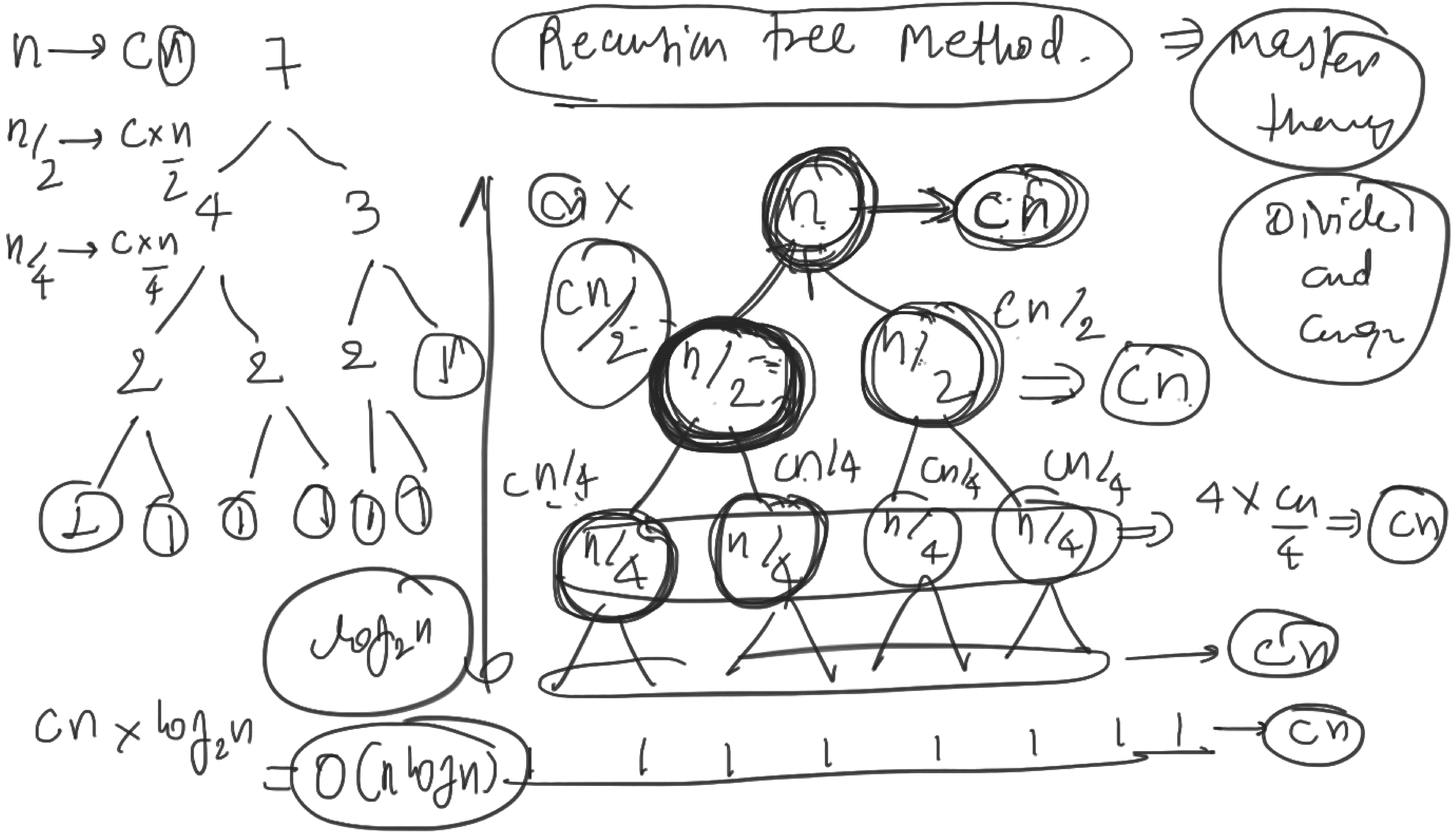
Base case ?

5 8 ⇒ $l == r$ $l > r$?

Recursion tree Method.

⇒ Master theory

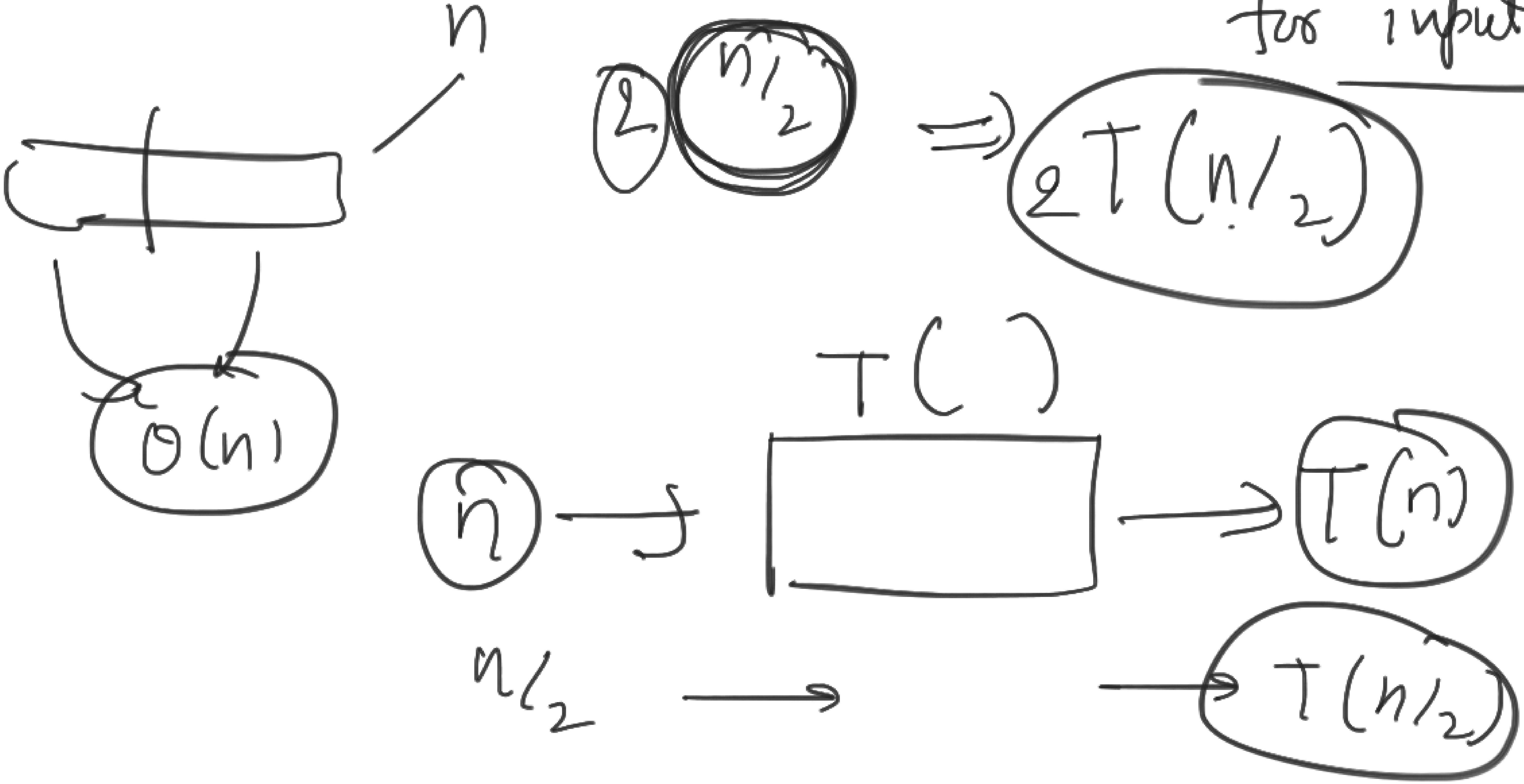
Divide and conqr



$cn \times \log_2 n = O(n \log n)$

Recurrence Relation-

$T(n) \Rightarrow$ time complexity for input size n



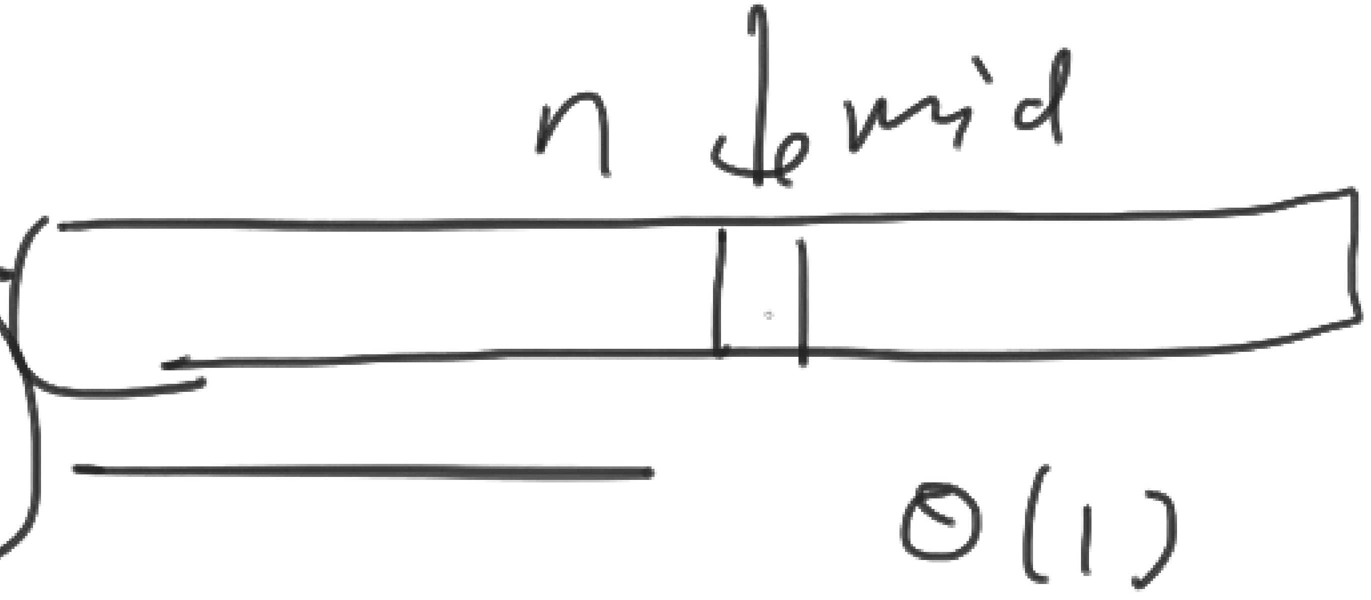
$$\Rightarrow T(n) = 2T(n/2) + O(n)$$

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

recurrence

Binary search

$$\Rightarrow T(n) = T(n/2) + O(1)$$



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

$$\boxed{fact(n) = fact(n-1) \times n} \Rightarrow \boxed{T(n) = T(n-1) + c}$$

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

$$= 2\left[2T(n/4) + \frac{cn}{2}\right] + cn$$

$$= 4T(n/4) + cn + cn$$

$$2^3 \cdot T(n/2^3) + \boxed{3 \cdot cn}$$

$$4\left(2T(n/8) + cn/4\right) + 2cn$$

$$8T(n/8) + cn + 2cn \Rightarrow \boxed{8T(n/8) + 3cn}$$

$$T(n) = 2^{k} \cdot T\left(n/_{2^{k}}\right) + \boxed{K} \cdot (cn)$$

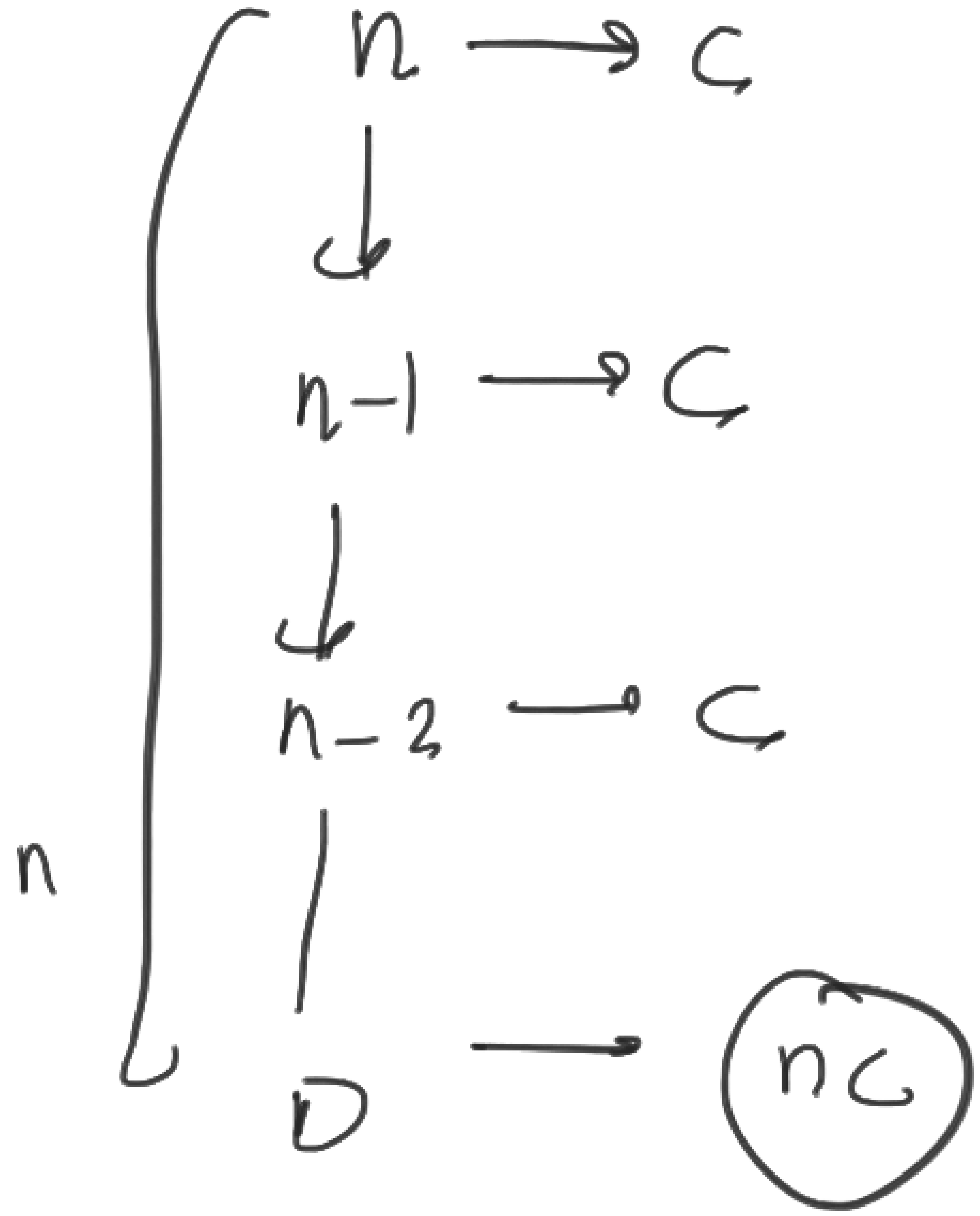
$$= \boxed{2^{\log n} \cdot T(1)} + \left(\log n \cdot cn\right)$$

$$\boxed{n/_{2^{k}} = 1} \Rightarrow K = \boxed{\log n}$$

$$= n + c\left(n \cdot \log n\right)$$

$$= \boxed{\Theta(n \log n)}$$

$$T(n) = T(n-1) + C \Rightarrow T(n) = O(n)$$



$$T(n) = T(n-1) + T(n-2) + C$$

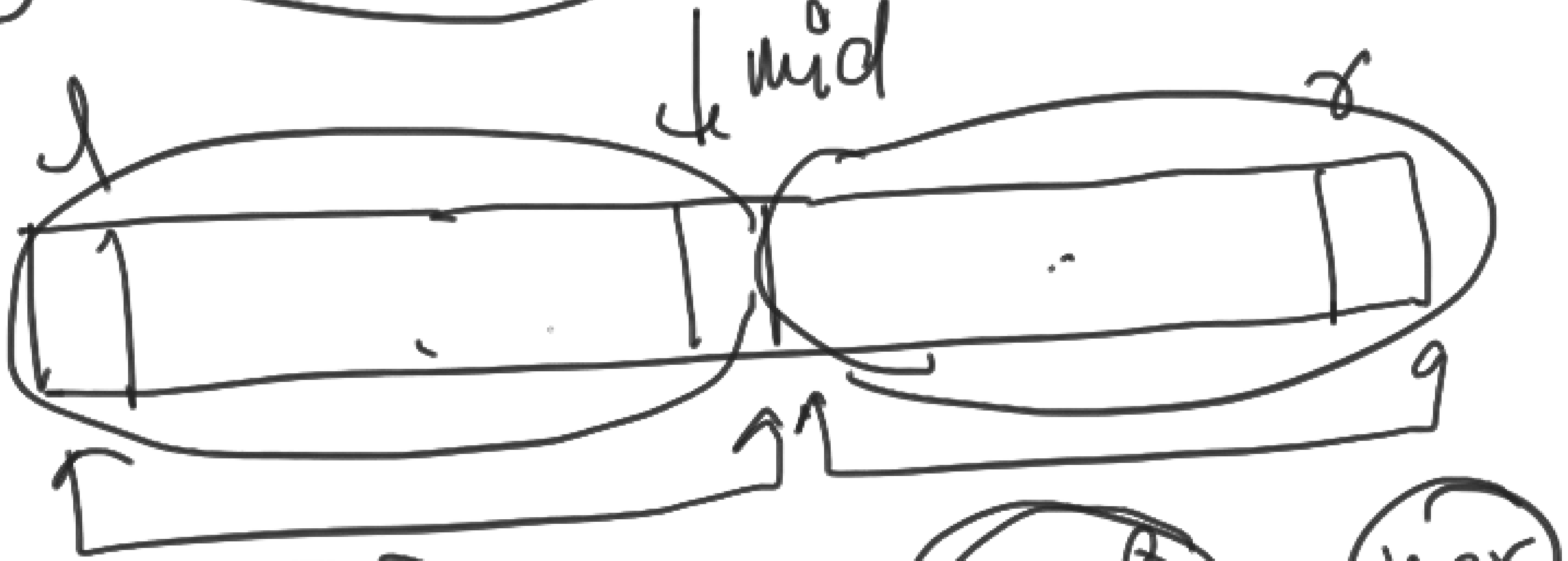
$$\Downarrow$$

$$T(n) = O(2^n)$$

find max and min.

Recursion

n



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

$+\ O(1)$

$$\boxed{T(n) = 2T(n/2) + c}$$

maxL

minL

2

maxR

minR

max

min

comp

$T(n) = ?$

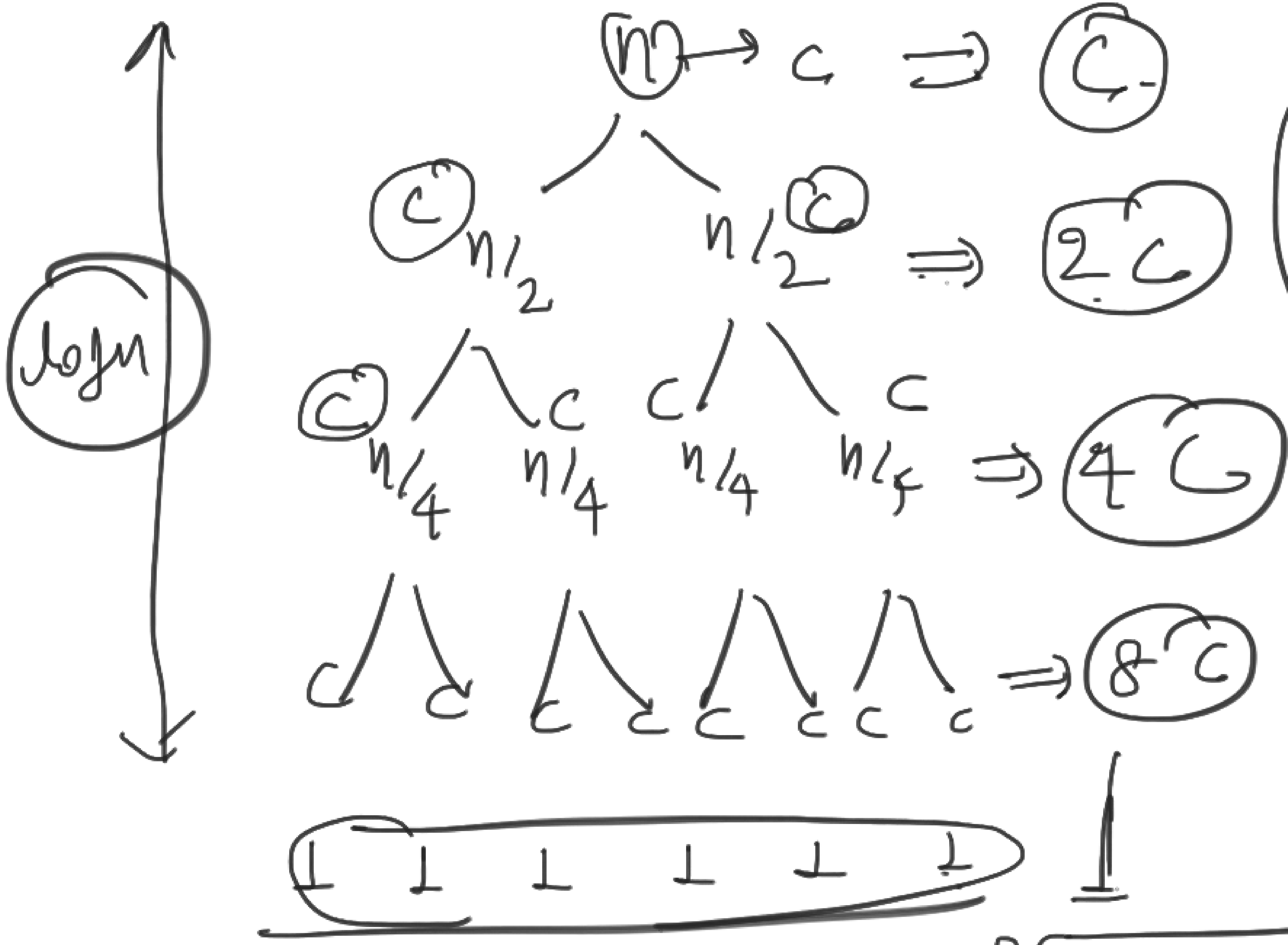
2

$O(\log n)$?

$O(n)$ ✓

$O(\log n)$

$$\boxed{T(n) = T(n/2) + c}$$



$$c + 2^1 c + 2^2 c + 2^3 c + \;-\;- + 2^{\log n} \cdot c \quad 2^{\log n + 1} - 1$$

$$1 + 2 + 2^2 \;-\;- n \Rightarrow 2^{n+1} - 1$$

$$c \left[1 + 2 + 2^2 + \;-\;- \log n \text{ times} \right]$$

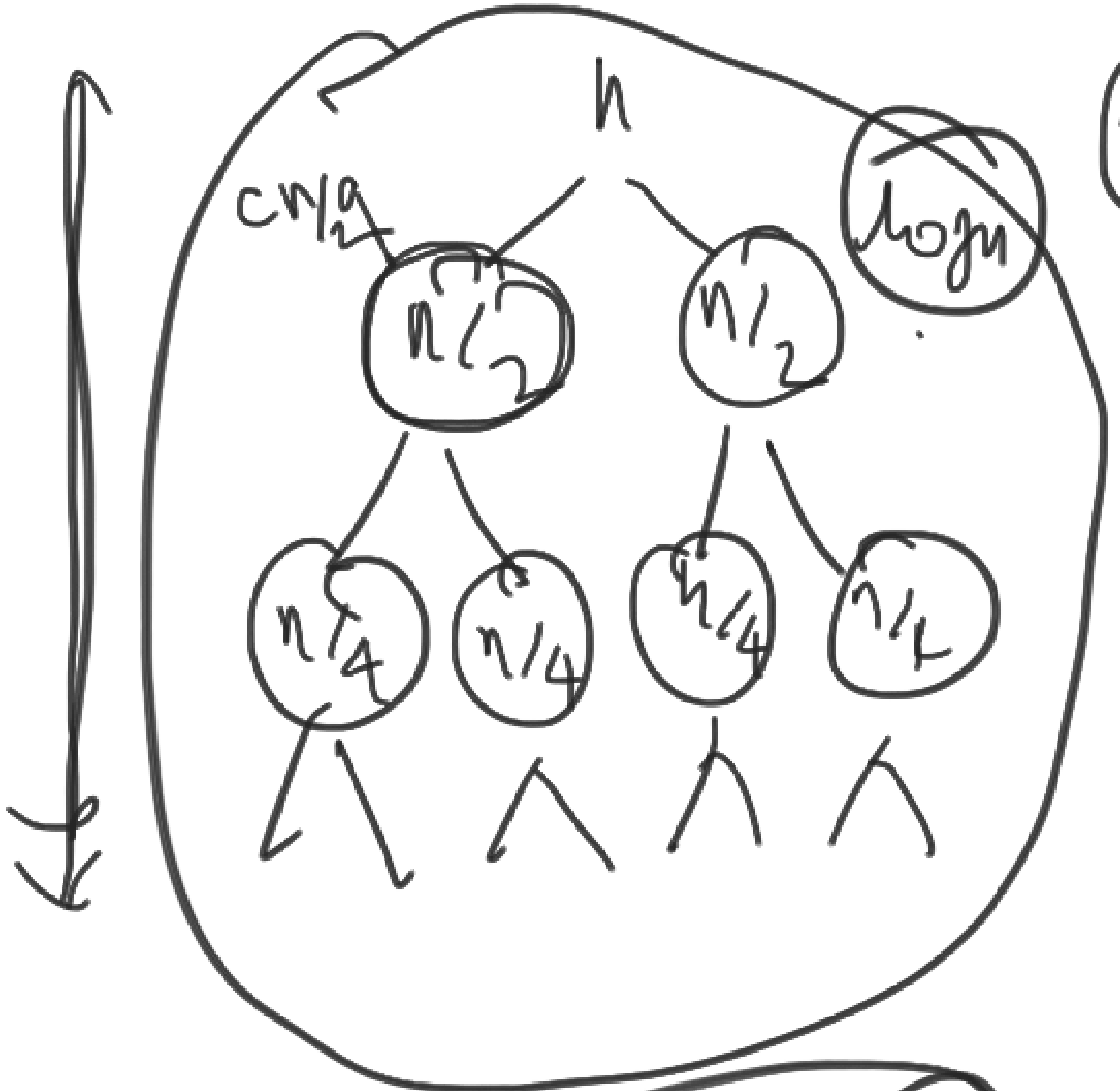
$$c\left[1 + 2 + 2^2 + 2^3 - - - \log n \text{ tems}\right]$$

$$= c\left[2^{\log n + 1} - 1\right]$$

$$\textcircled{a}^{\log_{\textcircled{a}} n} = \textcircled{n}$$

$$= c\left[2 \cdot \overset{n}{\textcircled{$2^{\log_2 n}$}} - 1\right]$$

$$= c\left[2n - 1\right] \overset{n}{=} \boxed{o(n)}$$



n Recursive calls

$\Rightarrow O(n \log n)$

$n \rightarrow 1$

$n/2 \rightarrow 2$

$n/4 \rightarrow \boxed{4}$

?

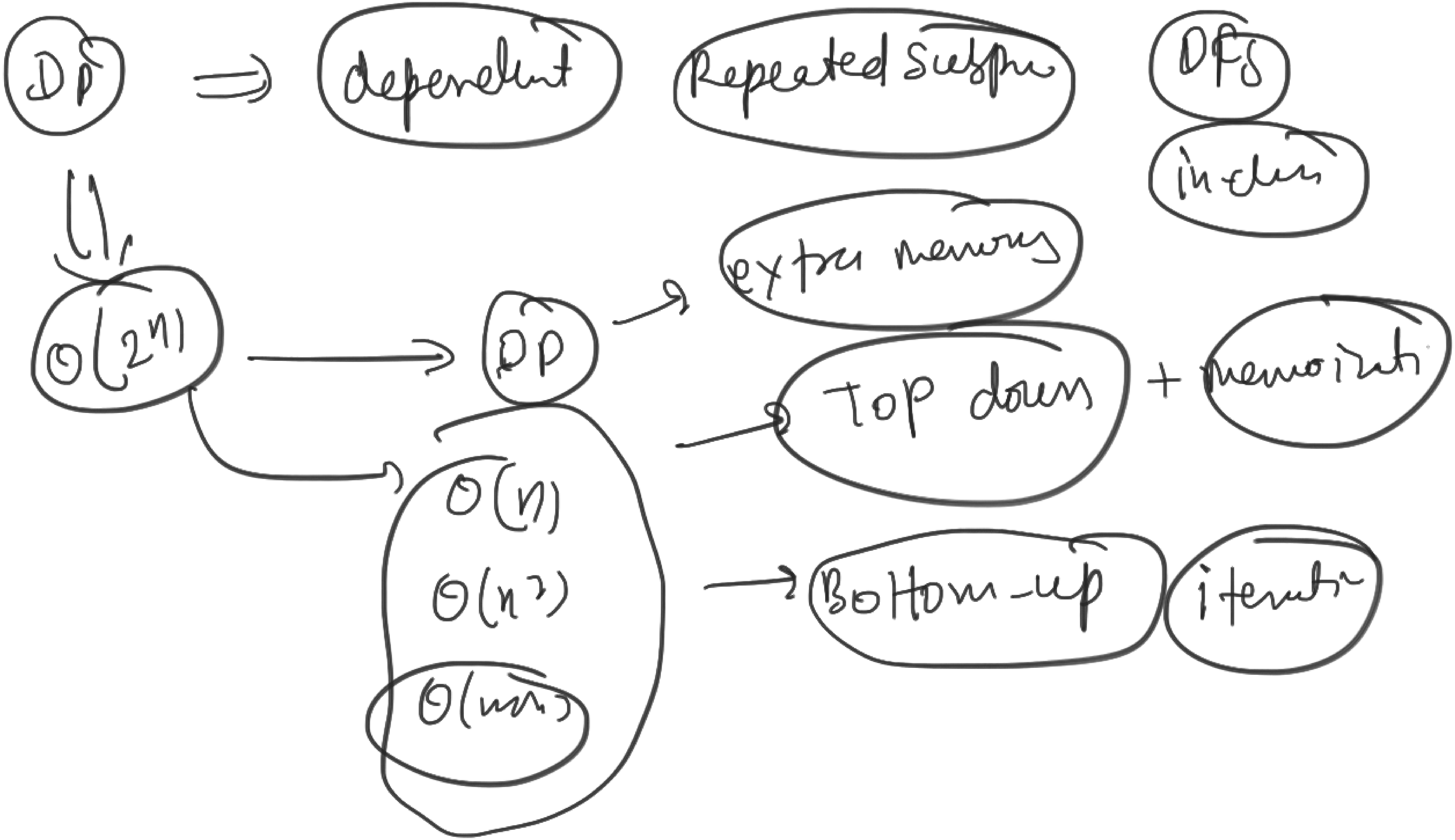
$O(2^n)$

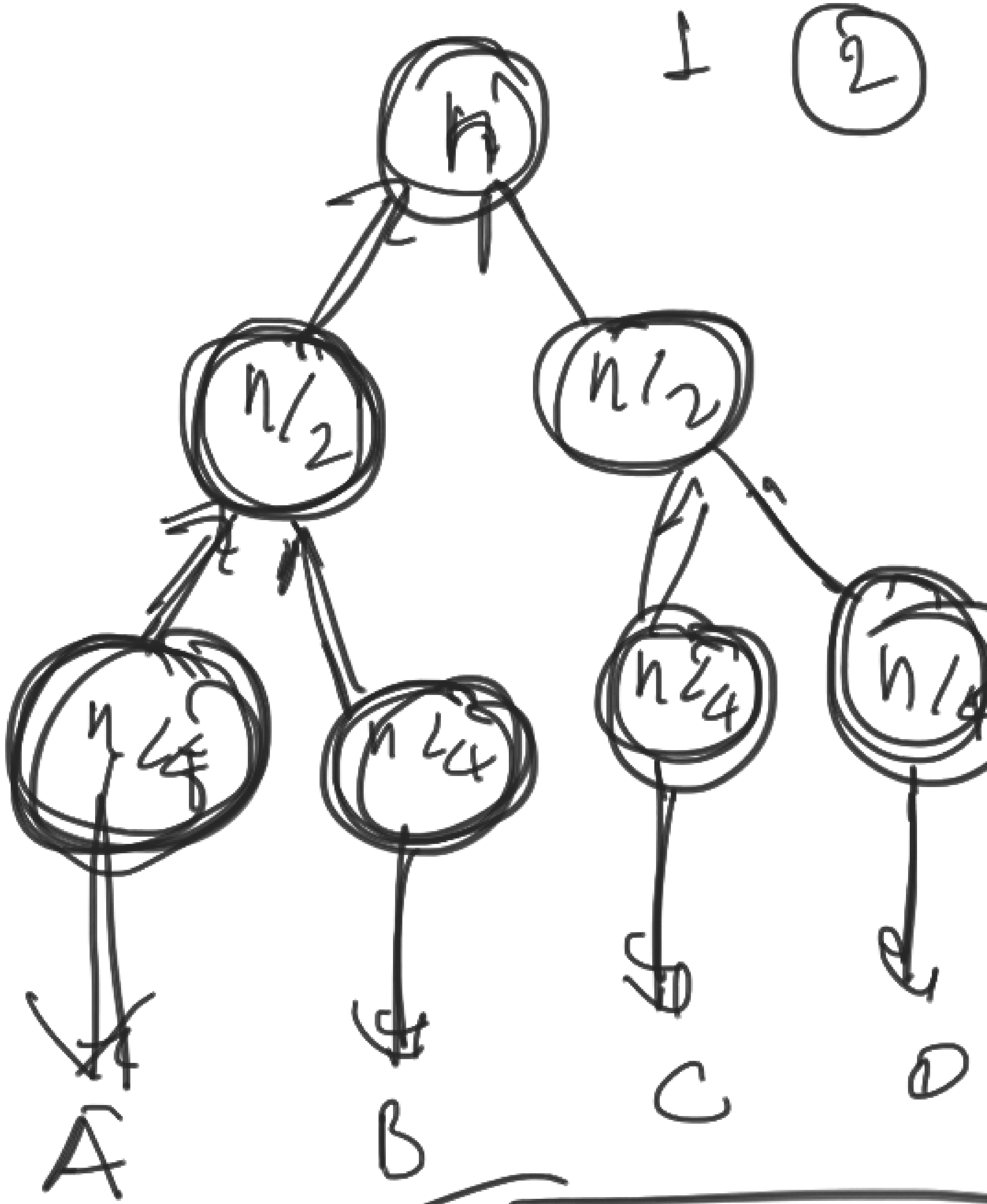
Subproblems are independent

$T(n) = 2T(n/2) + \boxed{Cn}$

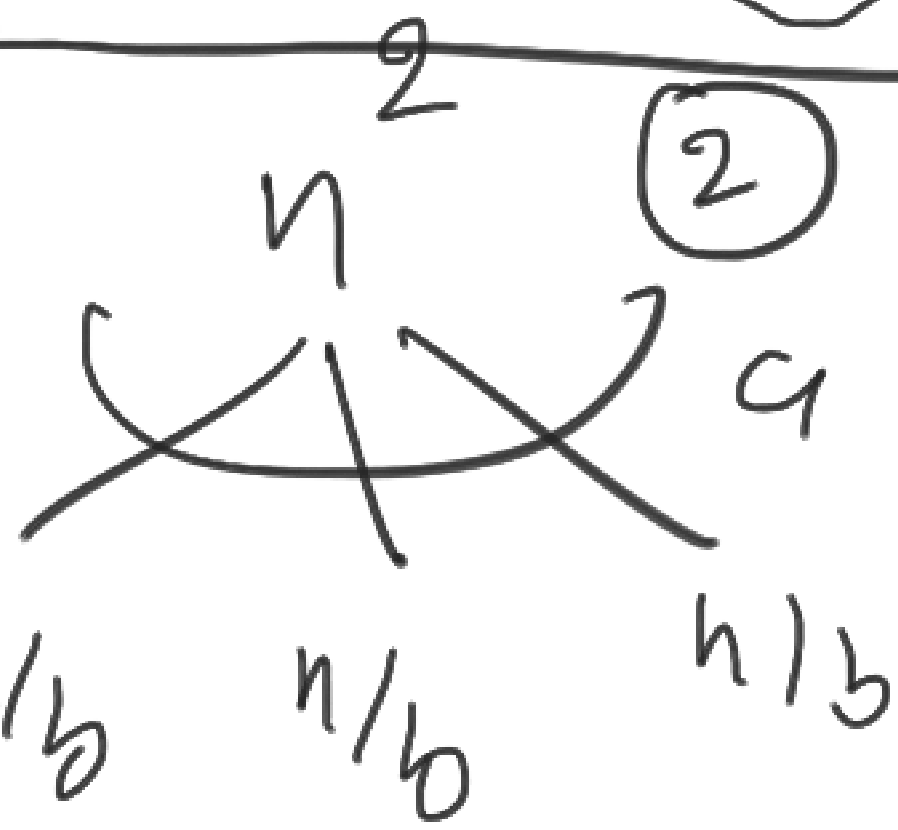
$T(n) = T(n-1) + T(n-2) + \boxed{C}$

n


DP
dependent
Repeated Subpro
DFS
inclus
O(2^n)
DP
extra memory
Top down
+
memoizati
Bottom-up
iterati
O(n)
O(n^2)
O(nm)



$$\boxed{T(n) = aT(n/b) + f(n)}$$



Master therm

extra

- $O(n)$
- $O(1)$
- $O(\log n)$
- $O(n^2)$

$$T(n) = T(n-1) + T(n-2) + C$$

$T(n)$?

⇒ Recursive insertion sort

⇒ Reverse array (recursion)

⇒ (Majority element (Divide and conqu)

⇒ nth Power ( )

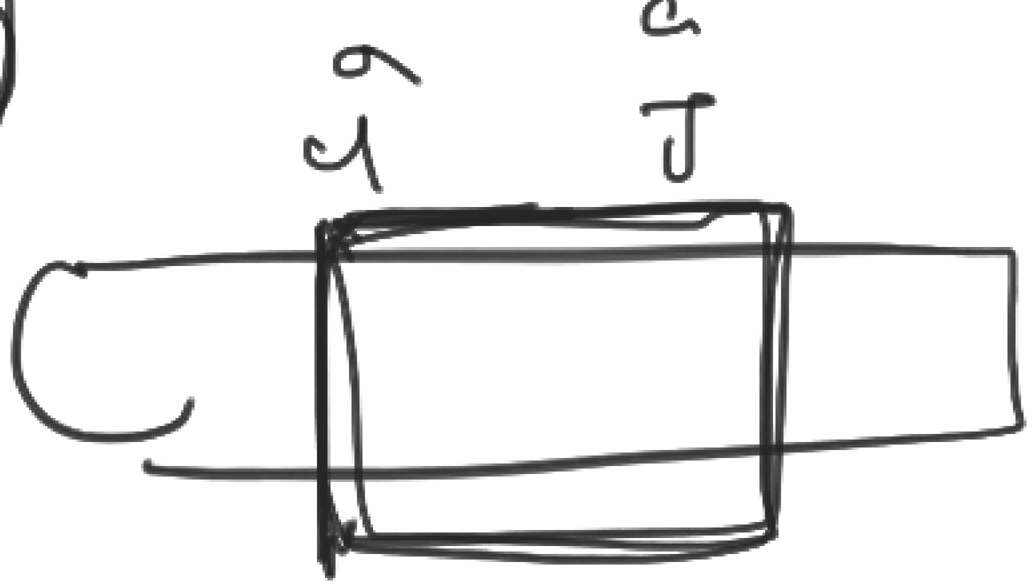
⇒ (max Subarray sum (Divide)

⇒ (find max diff array ( )

⇒ GCD ⇒

x , n

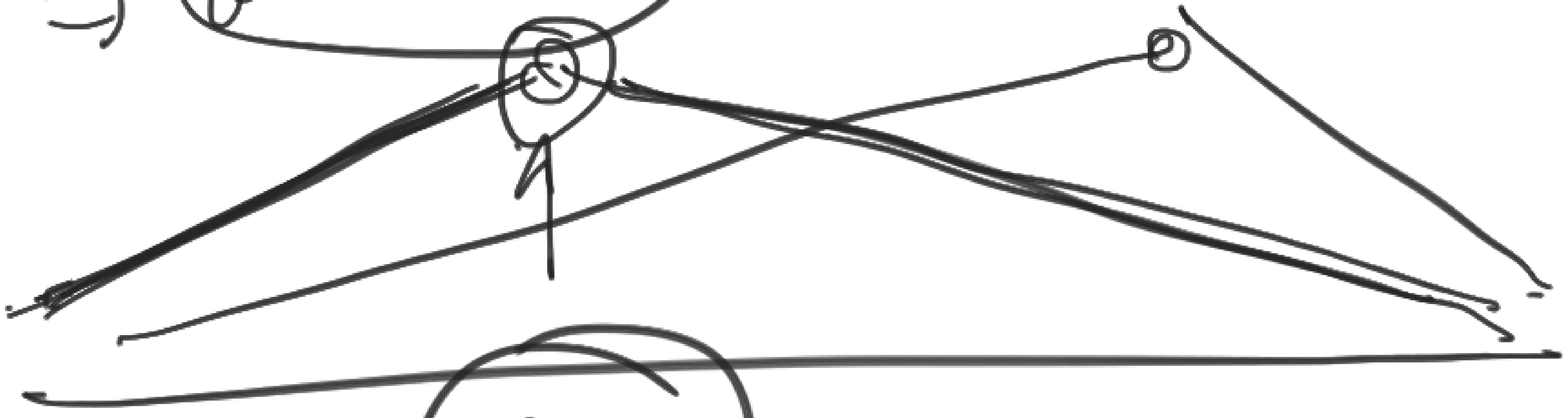
x^n



$$\frac{n(n+1)}{2}$$

$$\max(A[j] - A[i])$$

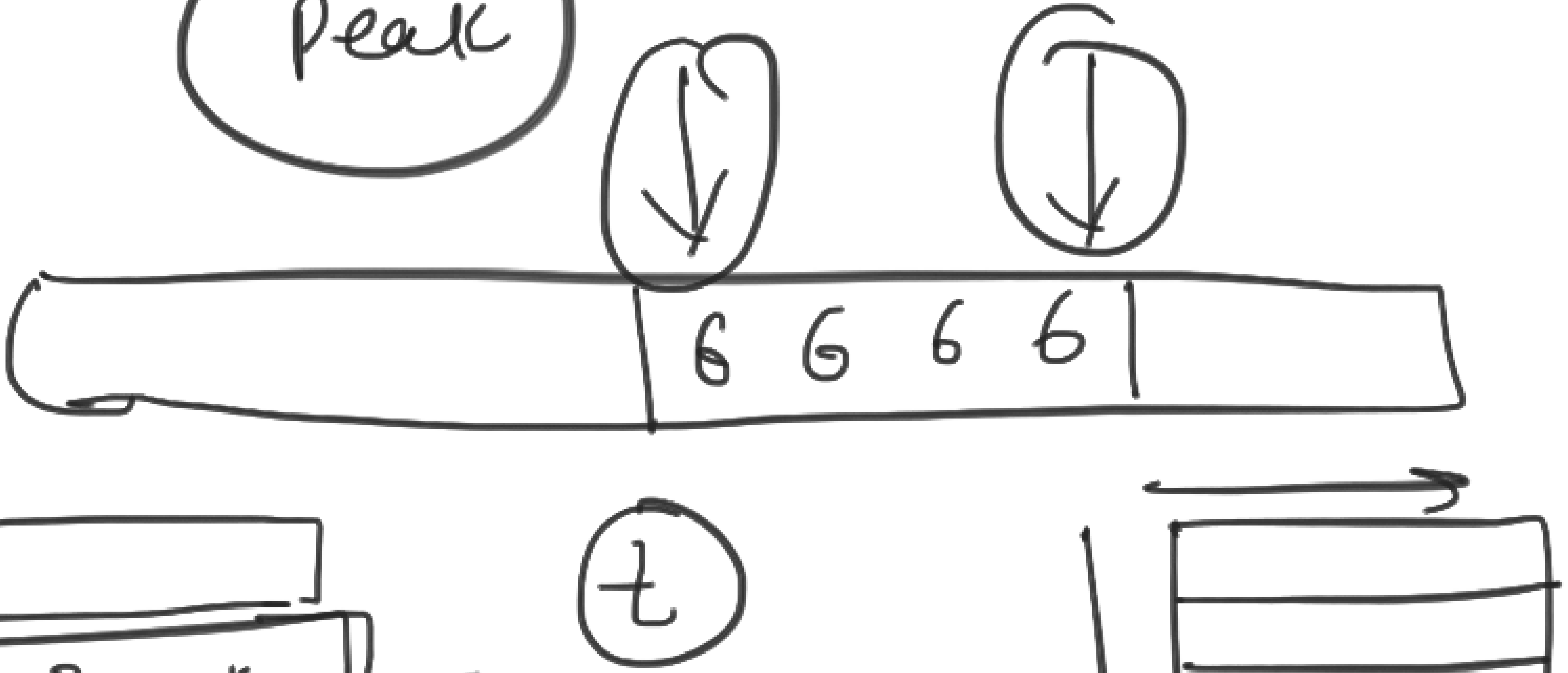
Binary search ⇒ quick sort



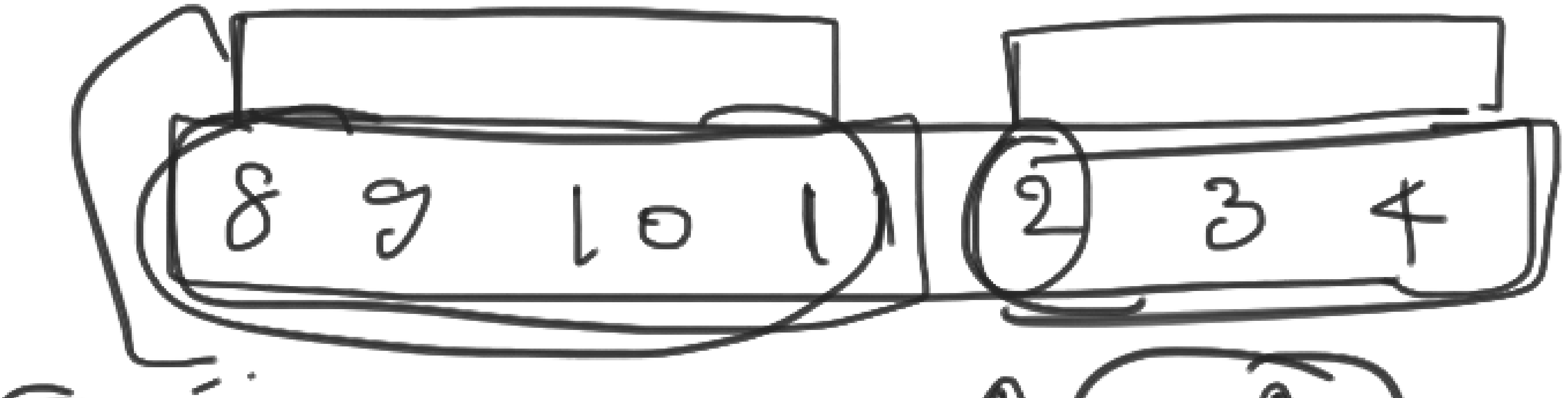
Peak

→

→



→ Sorted and Rotated



search t

min

target