$$T(n) = \begin{cases} 1 \; ; n = 1 \\ 8T(n/2) + n^2 \; \text{if } n > 1 \end{cases}$$

Do it in master theorem method in exam.

Here we do substitution method for practice

$$T(n) = 8\,T(n/2) + n^2 \rightarrow (1)$$

$$T(n/2) = 8\,T(n/4) + \left(\frac{n}{2}\right)^2 \rightarrow (2)$$

$$T(n/4) = 8\,T(n/8) + \left(\frac{n}{4}\right)^2 \rightarrow (3)$$

Subs (2) in (1)

$$T(n) = 8\left(8\,T(n/4) + \left(\frac{n}{2}\right)^2\right) + n^2$$

$$= 8^2\,T(n/2^2) + 2n^2 + n^2 \rightarrow (4)$$

Sub (3) in 4

$$8^2 T(n/2^2) + 2n^2 + n^2$$

$$= 8^2\left(8\,T\left(\frac{n}{8}\right) + \left(\frac{n}{4}\right)^2\right) + 2n^2 + n^2$$

$$= 8^3 T(n/2^3) + 2^2 n^2 + 2n^2 + n^2$$

$$\Big\}\ T \text{ times}$$

$$= 8^k T(n/2^k) + 2^{k-1} n^2 + 2^{k-2} n^2 + \cdots + 2^2 n^2 + 2n^2 + n^2$$

$$= 8^k T(n/2^k) + n^2\left(2^{k-1} + 2^{k-2} + \cdots + 2^2 + 2^1 + 2^0\right)$$

$$8^k T(n/2^k) + n^2(2^{k-1} + 2^{k-2} + \cdots + 2^2 + 2^1 + 2^0)$$

$$\text{Let } \frac{n}{2^k} = 1 \Rightarrow n = 2^k \Rightarrow k = \log n \,, \quad T\left(\frac{n}{2^k}\right) = 1 \text{ if } \frac{n}{2^k} = 1$$

$$a^{\log_c b} = b^{\log_c a}$$

$$8^{\log_2 n} + n^2(\underbrace{2^0 + 2^1 + 2^2 + \cdots 2^{\log n - 1}}_{GP = a\left(\frac{r^{n}-1}{r-1}\right)}) \qquad r > 1$$

$$a = 1, \quad r = 2, \quad n \rightarrow \log n$$

$$n^{\log_2 8}$$

$$n^3 + n^2\left(\frac{2^{\log_2 n} - 1}{2-1}\right) \qquad 2^{\log_2 n} \rightarrow n^{\log_2 2^2} \rightarrow n$$

$$n^3 + n^2\left(\frac{n-1}{2-1}\right) = n^3 + n^3 = O(n^3)$$

$$= \Theta(n^3)$$

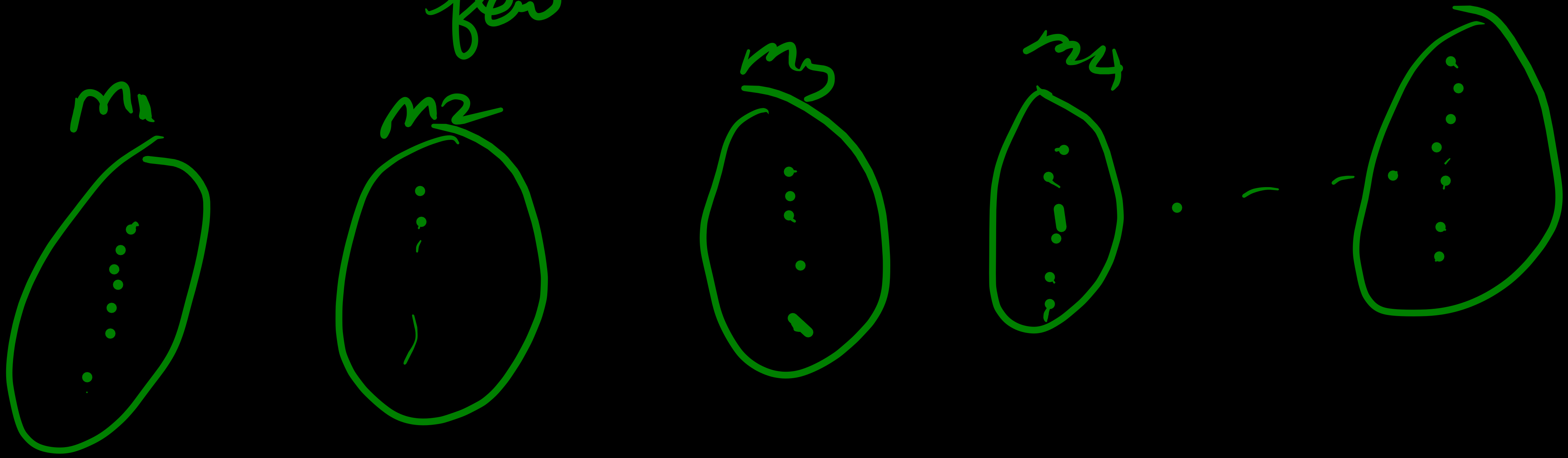
$$= \Omega(n^3)$$

Internet → models

Text books → model

CLRS 1 question.

few exercise quest.



$$T(n) = \begin{cases} 2 & \text{if } n=2 \\ 7\,T(n/2) + n^2 , & n>2 \end{cases}$$

$$T(n) = 7\,T(n/2) + n^2$$

$$= 7\left(7\,T(n/2^2) + (n/2)^2\right) + n^2$$

$$= 7^2\,T(n/2^2) + 7/4\, n^2 + n^2$$

$$= 7^2\left[7\,T(n/2^3) + \left(\frac{n}{2^2}\right)^2\right] + 7/4\, n^2 + n^2$$

$$= 7^3\,T(n/2^3) + (7/4)^2 n^2 + (7/4) n^2 + n^2$$

$$7^3 T(n/2^3) + (7/4)^2 n^2 + (7/4) n^2 + n^2$$

$\Big\{$ k times

$$= 7^{k} T(n/2^k) + (7/4)^{k-1} n^2 + (7/4)^{k-2} n^2 + \cdots\cdots (7/4)^2 n^2 + 7/4\, n^2 + n^2$$

$$n/2^k = 2 \Rightarrow n = 2^{k+1} \Rightarrow \log n = k+1 \Rightarrow k = \log n - 1$$

$$= 7^{\log n - 1}(2) + n^2\left((7/4)^0 + (7/4)^1 + (7/4)^2 + \cdots\cdots + (7/4)^{\log n - 2}\right)$$

$$= \frac{7^{\log_2 n}}{7} \cdot 2 + n^2\left(1 \cdot \frac{(7/4)^{\log n - 1} - 1}{7/4 - 1}\right)$$

$$= \frac{7^{\log_2 n}}{7} \cdot 2 + n^2 \left(\frac{\left(\frac{7}{4}\right)^{\log n - 1} - 1}{\frac{7}{4} - 1} \right)$$

$$= n^{\log_2 7} + n^2 \left(\frac{7}{4}\right)^{\log n}$$

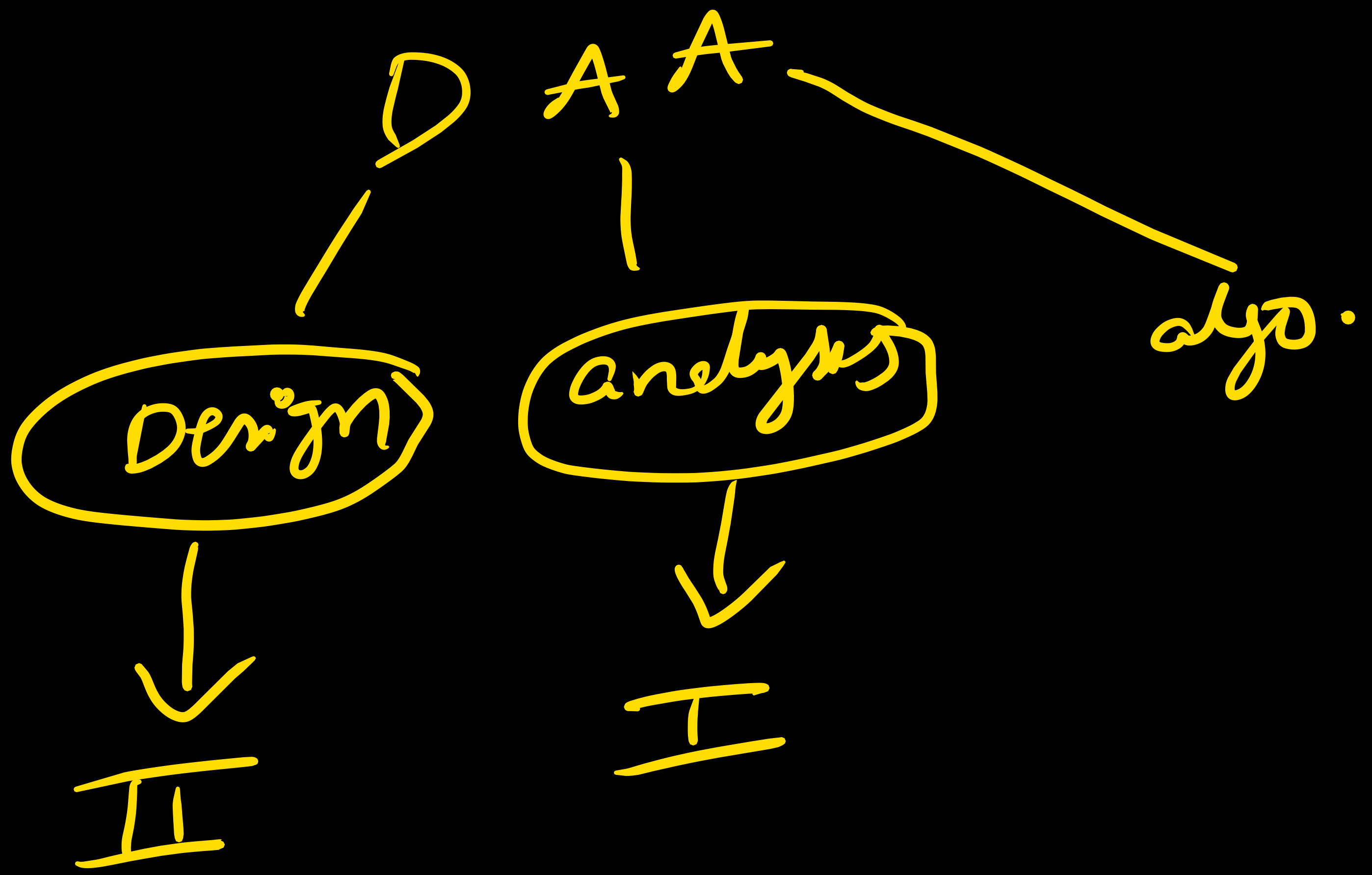
$$= \underline{\underline{n^{\log_2 7}}} + \cancel{n^2} \frac{7^{\log_2 n}}{\cancel{n^2}\, 4^{\log_2 n}} = n^{\log_2 7}$$

$$4^{\log_2 n} \rightarrow n^{\log_2 4} = n^2.$$

$$= O(n^{2.81})$$

Every question we are solving here is some algorithm. ~~Lot of~~ Some of them are in syllabus. We will see them when the time comes.

analysing algo.



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

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

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

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

$$= 2^2\left(T(n/2^3) + n/4\right) + n + n$$

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

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

$\Big\}$ k time

$$2^k T(n/2^k) + k n$$

let $n/2^k = 1 \Rightarrow n = 2^k \Rightarrow k = \log_2 n$

$$2^{\log_2 n}(1) + (\log_2 n) n.$$

$$n^{\log_2 2} + (\log_2 n) n = O(n \log n)$$

$\Omega(n \log n)$

$\Theta(n \log n)$

*****

Q)

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

$$T(n) = 2T(n/2) + n\log n$$

$$= 2\left(2T(n/4) + n/2 \log n/2\right) + n\log n$$

$$= 2^2 T(n/2^2) + 2n\log n/2 + n\log n$$

$$= 2^2 T(n/2^2) + 2n\log n/2 + n\log n$$

$$= 2^2\left[T(n/8) + \frac{n}{2^2}\log\frac{n}{2^2}\right] + n\log\frac{n}{2} + n\log n$$

$$= 2^3 T(n/2^3) + n\log\frac{n}{2^2} + n\log\frac{n}{2} + n\log\frac{n}{2^0}$$

k times

$$2^k T(n/2^3) + n\log\frac{n}{2^{k-1}} + n\log\frac{n}{2^{k-2}} + \cdots + n\log n$$

$$2^k\, T\left(n/2^k\right) + n\log\frac{n}{2^{k-1}} + n\log\frac{n}{2^{k-2}} + \cdots + \cancel{n\log n} \quad n\log n$$

$$= \cancel{n}\, 1$$

let $\frac{n}{2^k} = 1 \Rightarrow 2^k = n \Rightarrow k = \log n.$

$$= n(1) + n\left(\log\frac{n}{2^0} + \log\frac{n}{2^1} + \cdots\cdots + \log\frac{n}{2^{k-1}}\right) \qquad \log\frac{n}{2^{\log n - 1}}$$

$$= n + n\left(\log n - \log 2^0 + \log n - \log 2^1 + \cdots\cdots + \log n - \log 2^{\log n - 1}\right)$$

$$= n + n\left(\log n - \log 2^0 + \log n - \log 2^1 + \cdots + \log n - \log 2^{\log n - 1}\right)$$

$$= n + n\left(\log n \log n - \log_2 2^0 - \log_2 2^1 + \cdots \log_2 2^{\log n - 1}\right)$$

$$= n + n\left\{(\log n)^2 - \left(\underbrace{1 + 2 + 3 + \cdots \log n - 1}_{\text{Sum of I } \log n - 1 \text{ nums}}\right)\right.$$

$$= n + n\left((\log n)^2 - \frac{(\log n - 1)(\log n)}{2}\right)$$

$$\frac{(\log n)^2}{2}$$

$n(\log n)^2$ ✓

$n \log n$ ✗

$$= n + \frac{n(\log n)^2}{2}$$

all question

Substitution

$$T(n) = 2\underbrace{T(n-1)}_{\text{one terms.}} + K$$

Tree

$$T(n) = \underbrace{T(n-1) + T(n-2)}_{\text{more than one term}}$$

master

$$T(n) = \underbrace{aT(n/b)}_{\text{not for all que}} + K$$

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

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

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

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

$$= 2^2(2T(n-3)+(n-2)) + 2(n-1) + n$$

$$= 2^3 T(n-3) + 2^2(n-2) + 2^1(n-1) + n$$

$$= 2^3 T(n-3) + 2^2(n-2) + 2^1(n-1) + n$$

$$\begin{aligned} &T(n) \\ -&2T(n) \\ \hline -&T(n) \end{aligned}$$

$\Big\}$ k times

$$2^k T(n-k) + 2^{k-1}(n-(k-1)) + 2^{k-2}(n-(k-2)) + \cdots\cdots + 2^1(n-1) + 2^0(n-0)$$

$n-k=1 \Rightarrow k=n-1$

$$= 2^{n-1}T(1) + 2^{n-2}(n-(n-1-1)) + 2^{n-3}(n-(n-1-2)) + \cdots 2^1(n-1) + 2^0(n-0)$$

$$T(n) = 2^{n-1}\cdot 1 + 2^{n-2}\cdot 2 + 2^{n-3}\cdot 3 + \cdots\cdots + 2^1(n-1) + 2^0(n-0)$$

$\Downarrow$

$\underline{\underline{T(n)}}$ — Combination of A P and $\underline{\underline{\text{GP}}}$

$$T(n) = 2^0(n-0) + 2^1(n-1) + 2^2(n-2) + \cdots\cdots + 2^{n-3}(3) + 2^{n-2}(2) + 2^{n-1}(1)$$

$$-2 \times T(n) = \quad -2^1(n-0) - 2^2(n-1) \quad \cdots \quad -2^{n-3}(4) - 2^{n-2}(3) - 2^{n-1}(2) - 2^n(1)$$

$$-T(n) = 2^0(n) - \left(2^1 + 2^2 + 2^3 + \cdots + 2^{n-3} + 2^{n-2} + \cdots 2^n\right)$$

$$\begin{array}{r} T(n) \\ -2T(n) \\ \hline -T(n) \end{array}$$

$$-T(n) = n - \underbrace{\left[2^1 + 2^2 + 2^3 + \cdots\cdots 2^n\right]}$$

$$T(n) = \left[2^1 + 2^2 + 2^3 + \cdots 2^n\right] - n$$

$$= \frac{2(2^n - 1)}{2-1} - n = O(2^n) \quad \Theta(2^n) \quad \Omega(2^n)$$

$$T(n) = \underline{n^{1/2}\, T(n^{1/2}) + n}$$

$$= n^{1/2}\left[n^{1/4}\, T(n^{1/4}) + n^{1/2} \right] + n$$

$$= n^{3/4}\, T(n^{1/4}) + n + n$$

$$= n^{3/4}\left(n^{1/2^3}\, T(n^{1/2^3}) + n^{1/4} \right) + n + n$$

$$= n^{7/2^3}\, T(n^{1/2^3}) + 3n$$

$$= n^{\frac{2^3-1}{2^3}}\, T(n^{1/2^3}) + 3n$$

$$= n^{\frac{2^3-1}{2^3}} T\left(n^{1/2^3}\right) + 3n$$

$\Downarrow$ k times

$$n^{\frac{2^k-1}{2^k}} T\left(n^{1/2^k}\right) + kn$$

$$n^{1-\frac{1}{2^k}} T\left(n^{1/2^k}\right) + kn$$

$$\frac{n}{n^{1/2^k}} T\left(n^{1/2^k}\right) + kn$$

let $n^{1/2^k} = 2$ $\rightarrow$ termination Condition

$$\left(\frac{n}{2}\right)(2) + \log\log n\,(n) = n + n\log\log n$$

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

$\Theta(n\log\log n)$

$\Omega(n\log\log n)$

$n^{1/2^k} = 2$

$T\left(n^{1/2^k}\right)$ $\qquad n^{1/2^k} = 2$

k

$$\log n^{1/2^k} = \log 2$$

$$\frac{1}{2^k}\log n = 1 \Rightarrow 2^k = \log n$$

$$\Rightarrow k = \log\log n$$

NOW = 5 minutes to sol

10 minutes to solve.

Revise 4 times ✓

2 min

$\log n^{1/2^k} = \cancel{2} \log 2$

$\frac{1}{2^k} \log n = 1 \Rightarrow \log n = 2^k \Rightarrow \cancel{\log\log n =} \log 2^k$

$\log\log n = \log 2^k$

$\Rightarrow \log\log n = k.$

Recursive tree method:

$$T(n) = \underbrace{T(n-1) + T(n-2)}_{\text{more than two terms}} + n.$$

<u>ter</u>

$T(n) =$

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

You can apply masters theorem

But we have practice RT method

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

$$T(n)$$

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

Substitution.

$$T\left(\frac{7n}{8}\right) + T(n/8) + n.$$

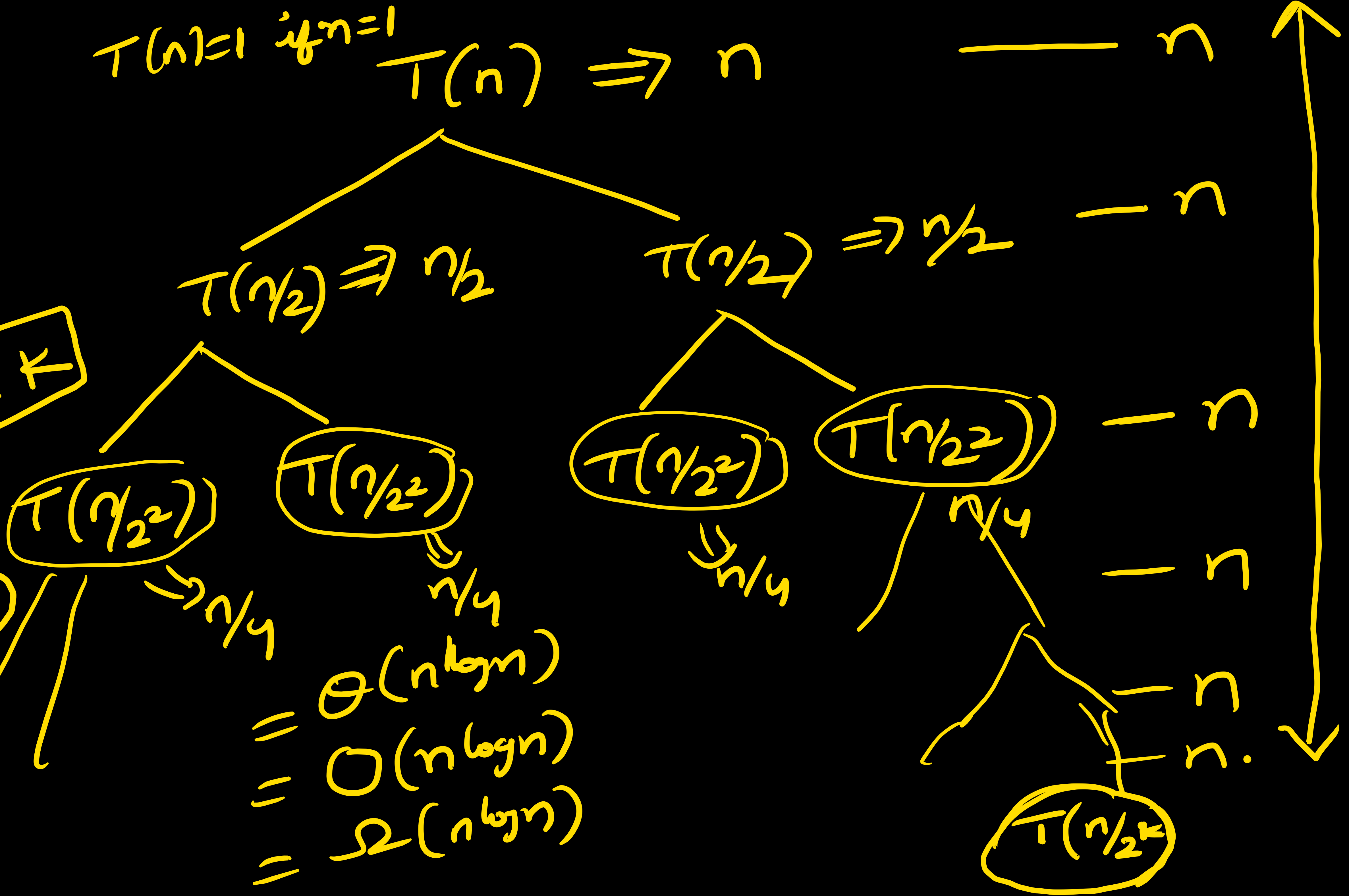
$$T(n) = T(n/2) + T(n/2) + n$$

Problem of size n

Problem of size $n/2$

Time taken for dividing problem into sub problems.

$T(n) = 1$ if $n = 1$



$T(n/2^k)$

$\downarrow$

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

$\Rightarrow \boxed{\log n = k}$

$n * \log n$

$= \Theta(n \log n)$

$= O(n \log n)$

$= \Omega(n \log n)$

assume

$\boxed{k \text{ levels}}$

SC = height of tree

$= k$

$= \log n$

$T(n) \Rightarrow n$ — n

$T(n/2) = n/2$ $\quad$ $T(n/2) = n/2$ — n

$T(n/2^2) = n/4$ $\quad$ $T(n/2^2) = n/4$ $\quad$ $T(n/2^2) = n/4$ $\quad$ $T(n/2^2) = n/4$ $= n$

⋮

$T(n/2^k)$

let us assume k level

log n

$n/2^k = 1 \Rightarrow n = 2^k \Rightarrow k = \log n$ ✓

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

$= 1$ if $n = 1$

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

$= n \times \log n$

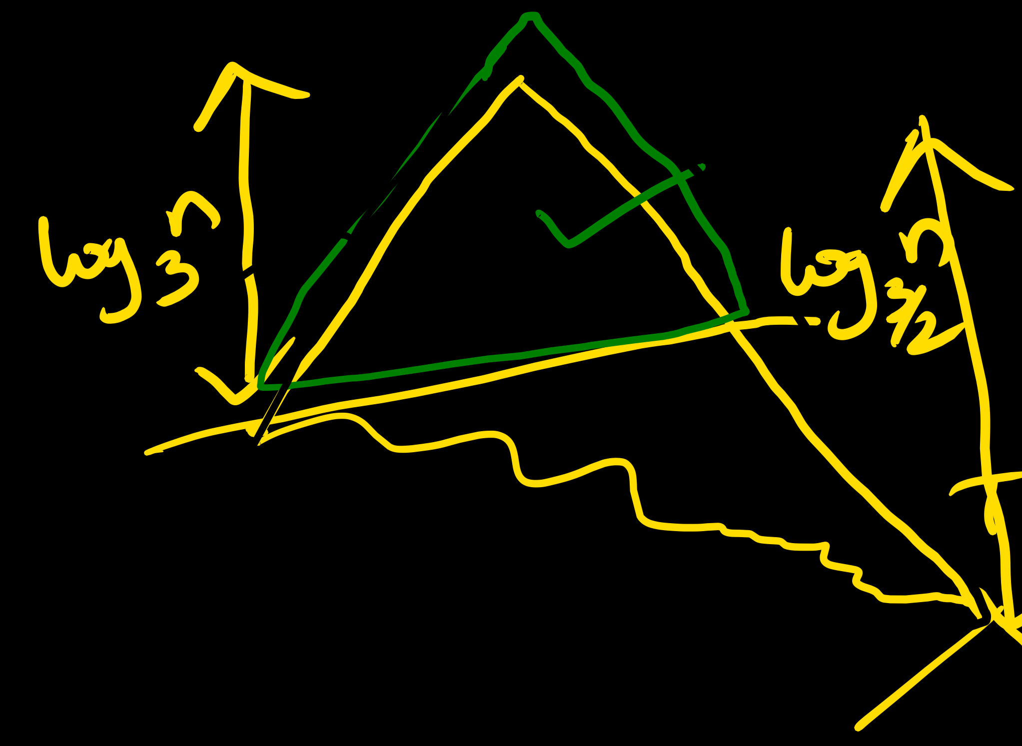
$O(n \log n)$

$\Omega(n \log n)$

$\Theta(n \log n)$

Gate 2016

$$T(n) = \begin{cases} 1 & \text{if } n = 1 \\ T(n/3) + T(2n/3) + n & \text{if } n > 1 \end{cases}$$



$T(n)$

$T(n/3)$ $T(2n/3)$

$T(n/3^2)$ $T\left(\frac{n}{(3/2)^2}\right)$