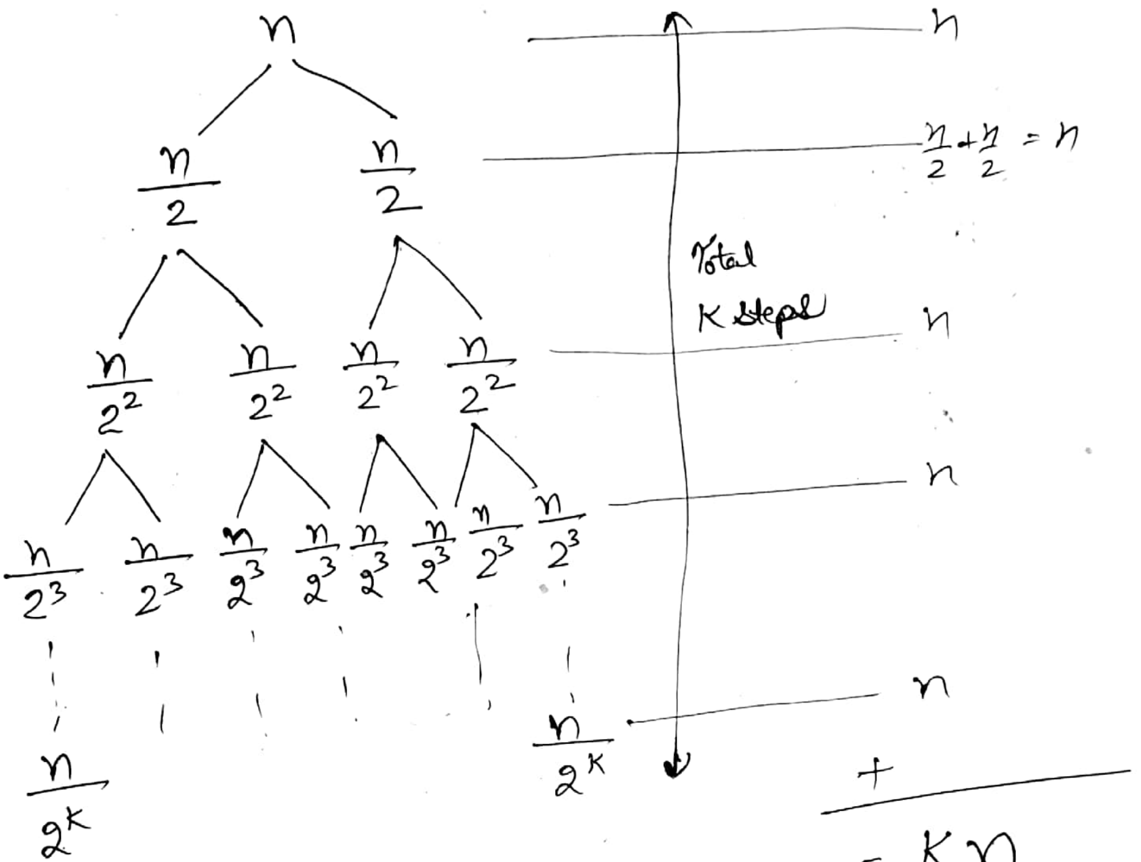
EXAMPLE OF RECURRENCE TREE METHOD :-

Example :-

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



$= Kn$

$= n \log n$

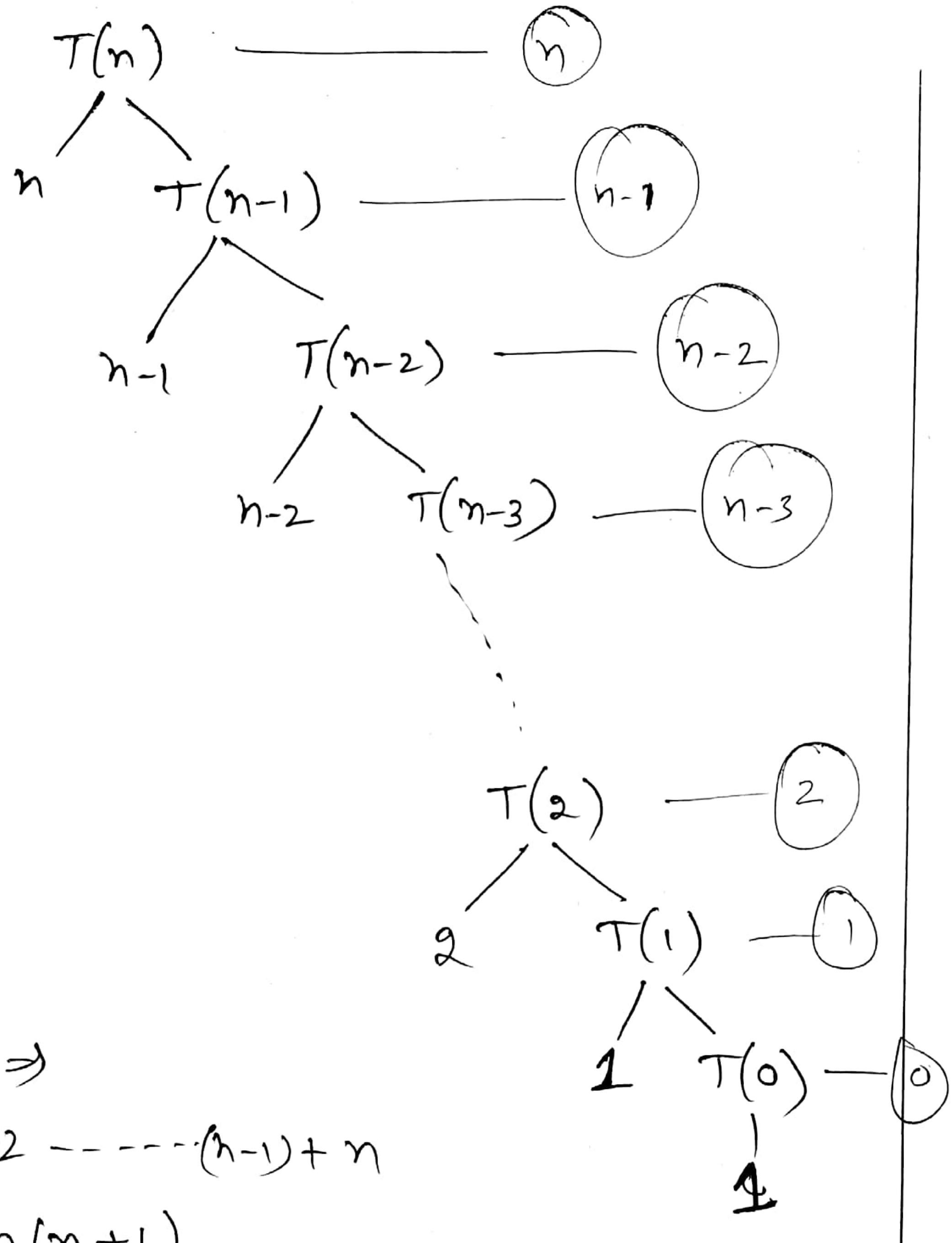
let assume $\frac{n}{2^k} = 1$

$\therefore n = 2^k$

$k = \log n$

$\boxed{O(n \log n)}$ Ans.

Example:- $T(n) = \begin{cases} 1 & , n=0 \\ T(n-1)+n & , n>0 \end{cases}$



Total time =>

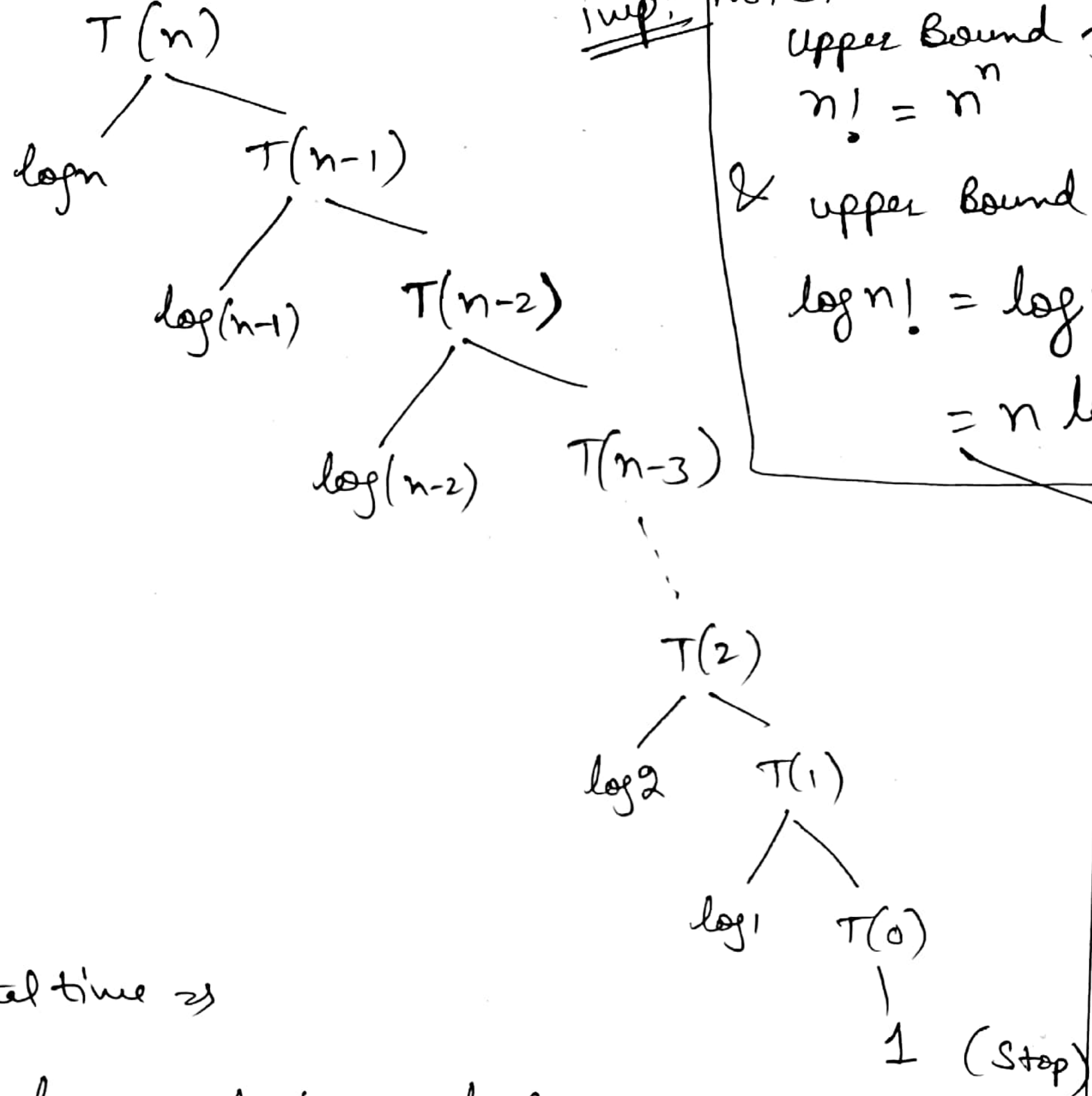
$\Rightarrow 0+1+2 \cdots\cdots (n-1)+n$

$\Rightarrow \dfrac{n(n+1)}{2}$

$\Rightarrow O(n^2)$

Example:-

$$T(n) = \begin{cases} 1 & , n = 0 \\ T(n-1) + \log n & , n > 0 \end{cases}$$



imp: **NOTE:-**

upper Bound for $n! = n^n$

& upper Bound for $\log n! = \log n^n$

$= n \log n$ — ①

Total time ⇒

$= \log n + \log(n-1) + \log(n-2) + \cdots \log 2 + \log 1$

$= \log[(n * (n-1) * (n-2) * \cdots 2 * 1]$

$= \log[n!]$ — No tight bound for this one But there is upper Bound for $\log n!$

using eq-①

$\therefore O(n \log n)$ Ans.

NOTE THIS :-

Every Time it will be multiply by n

$$T(n) = T(n-1) + 1 \text{ ——— } O(n) \quad (\text{i.e. } 1 \times n)$$

$$T(n) = T(n-1) + n \text{ ——— } O(n^2) \quad (n \times n)$$

$$T(n) = T(n-1) + \log n \text{ ——— } O(n \log n) \quad (n \times \log n)$$

$$T(n) = T(n-1) + n^2 \text{ ——— } O(n^3) \quad n \times n^2$$

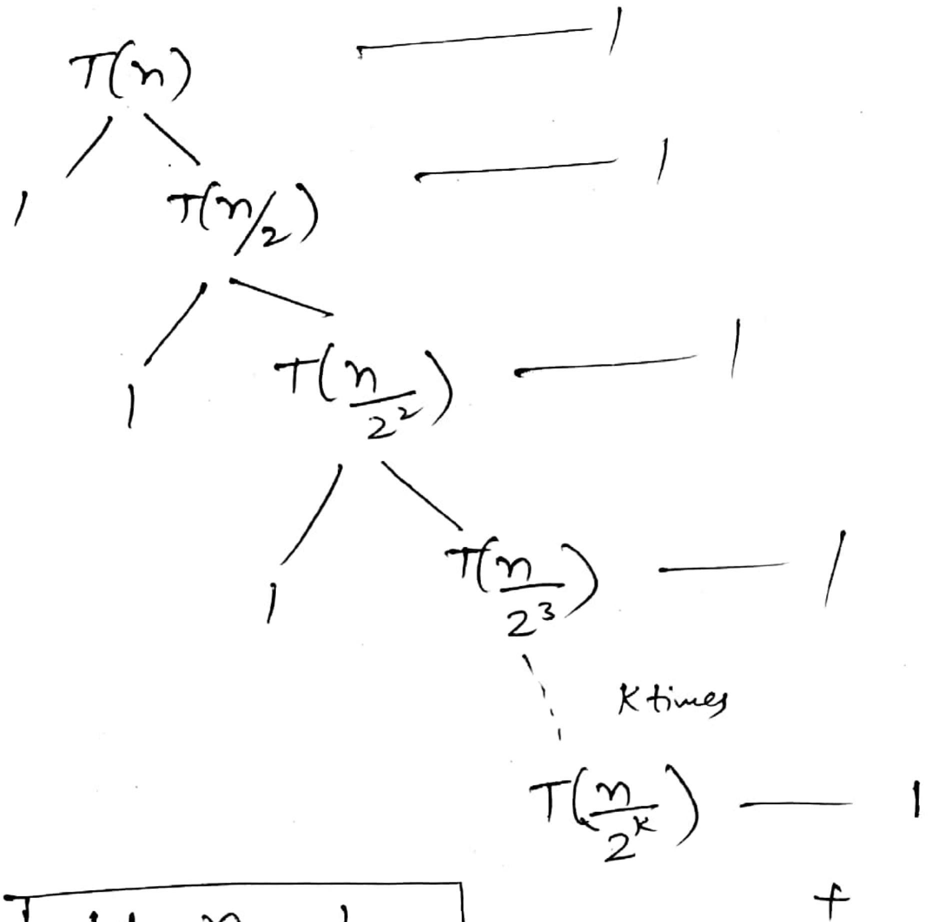
Now if we write it as

$$T(n) = T(n-2) + 1 \text{ ——— } \frac{n}{2} \Rightarrow O(n)$$

Illy $T(n) = T(n-100) + n$ ——— $O(n^2)$

imp:- When No Co-efficient is given with T $\longrightarrow$ Than whatever value is give just multiply it by (n)

Example:- $T(n) = \begin{cases} 1 & , n = 1 \\ T\left(\frac{n}{2}\right) + 1 & , n > 1 \end{cases}$



$$= \frac{}{1 \times K}$$

$\boxed{\begin{aligned} \text{let } \frac{n}{2^k} &= 1 \\ n &= 2^k \\ k &= \log n \end{aligned}}$

$$= 1 * K$$

$$= 1 * \log n$$

$$= \log n$$

$O(\log n)$ Ans.

Example :- $T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n$

n ——— n

$\frac{n}{3}$ $\quad$ $\frac{2n}{3}$ ——— $\frac{n}{3} + \frac{2n}{3} = \frac{\cancel{3}n}{\cancel{3}} = n$

$\frac{n}{3^2}$ $\quad$ $\frac{2n}{3^2}$ $\quad$ $\frac{2n}{3^2}$ $\quad$ $\frac{4n}{3^2}$ ——— $\frac{n}{9} + \frac{2n}{9} + \frac{2n}{9} + \frac{4n}{9} = \frac{\cancel{9}n}{\cancel{9}} = n$

$\frac{n}{3^3}$ $\quad$ $\frac{2n}{3^3}$ $\quad$ $\frac{2n}{3^3}$ $\quad$ $\frac{4n}{3^3}$ $\quad$ $\frac{2n}{3^3}$ $\quad$ $\frac{4n}{3^3}$ $\quad$ $\frac{4n}{3^3}$ $\quad$ $\frac{8n}{3^2}$ ——— $\frac{\cancel{27}n}{\cancel{27}} = n$

$\frac{n}{3^K}$

↑ Leftmost gives min. Time

imp

See Rightmost all Nodes

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

we write it as

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

$$\frac{4n}{3^2} = \frac{4n}{9} = \frac{n}{(3/2)^2}$$

$$\frac{8n}{3^3} = \frac{n}{\left(\frac{3}{2}\right)^3}$$

$\vdots$

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

Rightmost gives Maxi. Time

$\boxed{n/(3/2)^K}$ ——— n

$+$

$= n * K$

Left Subtree Height

let

$$\frac{n}{3^k} = 1$$

$$n = 3^k$$

$$k = \log_3 n$$

Right Subtree height.

let $$\frac{n}{\left(\frac{3}{2}\right)^k} = 1$$

$$n = \left(\frac{3}{2}\right)^k$$

$$k = \log_{3/2} n$$

NOTE=> ($\log_{3/2}$ is greater than $\log_3$ mathematically)

$$= n * k$$

↓ Put max. value.

$$= n * \log_{3/2} n$$

$$= n \log_{3/2} n \; .$$

Ans.

Max. Time

If min Time=>

$$= n * k$$

$$= n * \log_3 n$$

$$= n \log_3 n$$

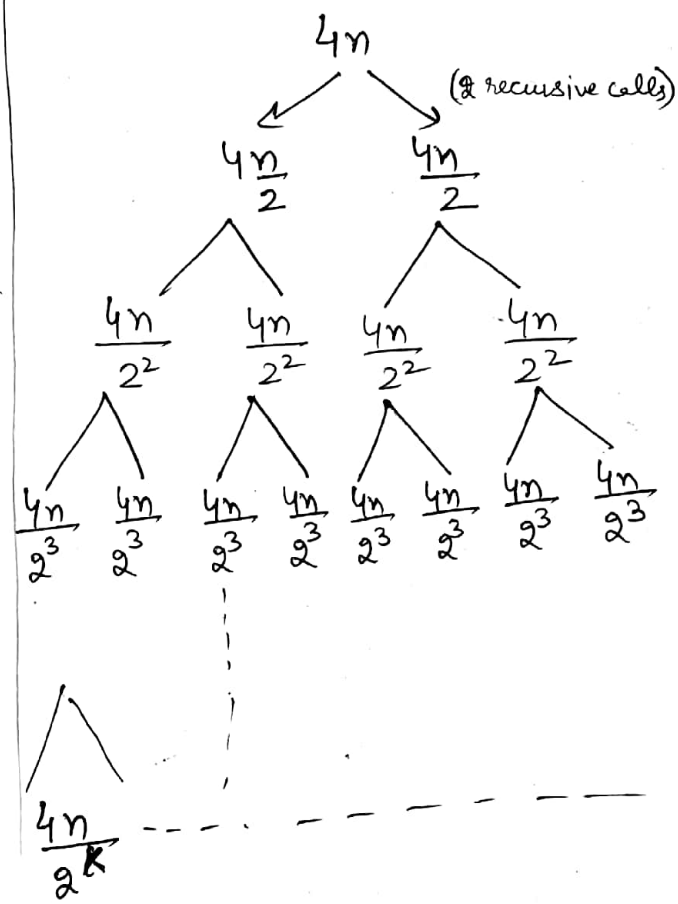
Scanned with CamScanner

Recurrence Tree Method :-

Example 1:- $T(n) = 2T\left(\frac{n}{2}\right) + 4n \quad , \; T(1) = 4$

Solⁿ →

Recursive Calls	No. of Nodes	TREE	Row Sum
$T(n)$	1	$4n$	$4n$
$T\left(\frac{n}{2}\right)$	2	$\frac{4n}{2}$	$4n$
$T\left(\frac{n}{2^2}\right)$	2^2	$\frac{4n}{2^2}$	$4n$
$T\left(\frac{n}{2^3}\right)$	2^3	$\frac{4n}{2^3}$	$4n$
⋮			⋮
$T\left(\frac{n}{2^k}\right)$	2^k	$\frac{4n}{2^k}$	$4n$



$$+ \quad \sum_{k=0}^{\log_2 n} 4n$$

→ Now Put this is equal to Base Case i.e. $T(1) = 4$

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

$$\log_2 n = k$$

$$= 4n \sum_{k=0}^{\log_2 n} 1$$

$$4n(\underbrace{1+1+1 \cdots 1}_{\log_2 n + 1})$$

$$= 4n(\log_2 n + 1)$$

$$= \boxed{4n\log_2 n + 4n}$$

Scanned with CamScanner