

1) Algorithm: $T(n) = (n-1)(n-2)(n-3) \dots 1$

Recursive relation: A problem is described by a mathematical relation, where a function is called again and again.

Substitution Method:

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

Solⁿ:

$$T(n) = n \cdot T(n-1)$$

$$T(n-1) = (n-1) \cdot T(n-2)$$

$$= (n-1) \cdot (n-2) \cdot T(n-3)$$

$$T(n-2) = (n-2) \cdot T(n-3)$$

$$= (n-2) \cdot (n-3) \cdot T(n-4)$$

From (1), (2) and (3), we get

$$T(n) = n \cdot (n-1) \cdot (n-2) \cdot (n-3) \cdot T(n-4)$$

$$\vdots$$

$$= n(n-1)(n-2)(n-3) \cdot T(n-(n-1))$$

$$= n(n-1)(n-2)(n-3) \dots 3.2.1$$

$$= n \cdot n \left(1 - \frac{1}{n}\right) \cdot n \left(1 - \frac{2}{n}\right) \cdot n \left(1 - \frac{3}{n}\right) \dots n \left(\frac{3}{n}\right) \cdot n \left(\frac{2}{n}\right) \cdot n \left(\frac{1}{n}\right)$$

2404022

$$= n^n \left[\left(1 - \frac{1}{5}\right) \left(1 - \frac{2}{5}\right) \left(1 - \frac{3}{5}\right) \cdots \left(\frac{3}{5}\right) \left(\frac{2}{5}\right) \left(\frac{1}{5}\right) \right]$$

∴ $T(n) = O(n^n)$. overhead, mit einer Laufzeit von n

10. Solve $T(n) = \begin{cases} 1 & \text{if } n=1 \\ 2T\left(\frac{n}{2}\right) + n & \text{if } n > 1 \end{cases}$

Soln:

$$T(n) = 2T\left(\frac{n}{2}\right) + n \quad \text{if } n > 1 \quad (1)$$

$$T\left(\frac{n}{2}\right) = 2T\left(\frac{n}{4}\right) + \frac{n}{2} \quad (2)$$

$$T\left(\frac{n}{4}\right) = 2T\left(\frac{n}{8}\right) + \frac{n}{4} \quad (3)$$

$$\therefore T(n) = 2T\left(\frac{n}{2}\right) + n \quad (1-2-N)T * (1-N) = (L-N)T$$

$$= 2 \left[2T\left(\frac{n}{4}\right) + \frac{n}{2} \right] + n \quad (L-N)T * (L-N) =$$

$$= 2^2 T\left(\frac{n}{4}\right) + (2n)T * (L-N) =$$

$$= 2^2 \left[2T\left(\frac{n}{8}\right) + \frac{n}{4} \right] + 2n \quad (L-N)T * (L-N) =$$

$$= 2^3 T\left(\frac{n}{8}\right) + 3n \quad (L-N)T * (L-N) + N = (N)T$$

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

$$(L-N) = 2^k T\left(\frac{n}{2^k}\right) + k n \quad (L-N)(L-N)N =$$

$$L \cdot S \cdot E \quad (L-N)(L-N)(L-N)N =$$

$$\left(\frac{L}{N}N \cdot \left(\frac{S}{N}N \cdot \left(\frac{E}{N}N \cdots \left(\frac{L}{N} - L\right)N \cdot \left(\frac{S}{N} - S\right)N \cdot \left(\frac{E}{N} - E\right)N\right)N\right)N =$$

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

$$\Rightarrow n = 2^k$$

$$\Rightarrow \log_2 n = \log_2 2^k$$

$$\Rightarrow k = \log_2 n$$

$$\therefore T(n) = n T\left(\frac{n}{2}\right) + n \log_2 n$$

$$= n \cdot 1 + n \log_2 n$$

$$= n + n \log_2 n$$

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

$$\text{Q. Solve } T(n) = \begin{cases} 1 & ; n=1 \\ T(n-1) + \log n & ; n>1 \end{cases}$$

Sol:

$$T(n) = T(n-1) + \log n \quad (1)$$

$$T(n-1) = T(n-2) + \log(n-1) \quad (2)$$

$$T(n-2) = T(n-3) + \log(n-2) \quad (3)$$

From (1) and (2),

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

$$= T(n-3) + \log(n-2) + \log(n-1) + \log n$$

$$\vdots$$

$$= T(n-k) + \log(n-(k-1)) + \log(n-(k-2)) + \dots + \log n$$

$$\text{Now, } n-k=1 \\ \Rightarrow k=n-1 \\ \approx n$$

$$\therefore T(n) = T(1) + \log_2 1 + \log_2 2 + \dots + \log_2 n \\ = 1 + \log_2 (1 \cdot 2 \cdot 3 \cdot \dots \cdot n) \\ = 1 + \log_2 (n!) \\ = 1 + \log_2 n^n [n! \approx n^n] \\ = 1 + n \log_2 n \\ \therefore T(n) = O(n \log n)$$

Q. Solve $T(n) = \begin{cases} T\left(\frac{n}{2}\right) + c & \text{if } n \geq 1 \\ 1 & \text{if } n=1 \end{cases}$

$$(1) T\left(\frac{n}{2}\right) + c + (n-1)P = (n)P$$

$$(2) T\left(\frac{n}{4}\right) + c + (n-2)P = (n)P$$

$$(3) T\left(\frac{n}{8}\right) + c + (n-3)P = (n)P$$

(Binary search equation)

Solⁿ: $T(n) = T\left(\frac{n}{2}\right) + c + (n-1)P = (n)P$

$$T\left(\frac{n}{2}\right) = T\left(\frac{n}{4}\right) + c \dots (2) \text{ base case}$$

$$T\left(\frac{n}{4}\right) = T\left(\frac{n}{8}\right) + c \dots (3) \text{ base case}$$

Substituting (2) in (1),

$$T(n) = T\left(\frac{n}{4}\right) + 2c + (n-2)P$$

$$= T\left(\frac{n}{8}\right) + 3c$$

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

and for $0 \leq k, 1 \leq d, 0 \leq c$

$$T\left(\frac{n}{2^k}\right) = (n)^D \text{ mult. } \frac{1}{2^k} \leq 0 \quad T(D)$$

$$= T\left(\frac{n}{2^k}\right) + kc$$

$$\text{let, } \frac{n}{2^k} = 1 \Rightarrow 2^k = n \Rightarrow (n)^D = (n)^D, D = \log_2(n)$$

$$\Rightarrow n = 2^k \Rightarrow (n)^D = (n)^D, D = \log_2(n)$$

$$\Rightarrow \log_2 n = \log_2 2^k \Rightarrow (n)^D = (n)^D, D = \log_2(n)$$

$$\Rightarrow k = \log_2 n \quad \text{mult. } 0 \leq k \leq D$$

$$\therefore T(n) = T(1) + c \log_2 n \quad \text{mult. } 0 \leq k \leq D$$

$$= 1 + c \log_2 n \quad (n)^D \text{ mult. } 0 \leq D$$

$$\therefore T(n) = O(\log_2 n)$$

$$T(n) = O(n^{\frac{1}{2}} + \log n) \quad 0 \leq q, D = \frac{1}{2}, k = \frac{1}{2}, A = 0$$

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

$$0 \leq k \leq D$$

$$0 \leq D \leq 1$$

$$(n)^D = (n^{\frac{1}{2}})^D = (n)^D \therefore$$

④ Master Method -

$$T(n) = aT\left(\frac{n}{b}\right) + \Theta(n^k \log^p n)$$

$a \geq 1, b \geq 1, k \geq 0$ and p is a real number.

(1) If $a > b^k$, then $T(n) = \Theta(n^{\log_b a})$

(2) If $a = b^k$,

$$(a) p > -1, T(n) = \Theta(n^{\log_b a} \cdot \log^{p+1} n) = \frac{C}{k!} n^{\log_b a} \cdot \log^{p+1} n$$

$$(b) p = -1, T(n) = \Theta(n^{\log_b a} \cdot \log \log n)$$

$$(c) p < -1, T(n) = \Theta(n^{\log_b a})$$

(3) If $a < b^k$,

$$(a) p \geq 0, T(n) = \Theta(n^k \log^p n)$$

$$(b) p < 0, T(n) = \Theta(n^k)$$

Ex:

$$(1) T(n) = 4T\left(\frac{n}{2}\right) + n$$

$$a = 4, b = 2, k = 1, p = 0$$

$$\text{Here, } a > b^k$$

$$\Rightarrow 4 > 2^1$$

$$\therefore T(n) = \Theta(n^{\log_2 4}) = \Theta(n^2).$$

$$(2) T(n) = T\left(\frac{n}{2}\right) + 1$$

$$a=1, b=2, k=0, p=0$$

$$\text{Hence, } a = b^k$$

$$\Rightarrow 1 = 2^0$$

$$\text{and } p > -1$$

$$\therefore T(n) = \Theta(n^{\log_2 1} \cdot \log n)$$

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

$$(3) T(n) = 2T\left(\frac{n}{2}\right) + \frac{n}{\log n}$$

$$a=2, b=2, k=1, p=-1$$

$$\text{Hence, } a = b^k$$

$$\Rightarrow 2 = 2^1$$

$$\text{and } p = -1$$

$$\therefore T(n) = \Theta(n^{\log_2 2} \cdot \log \log n)$$

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

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

$$a=2, b=2, k=1, p=-2$$

$$\text{Hence, } a = b^k$$

$$\Rightarrow 2 = 2^1$$

$$\text{and } p < -1$$

$$0 > b^k$$

$$\therefore T(n) = \Theta(n^{\log_2 1})$$

$$= \Theta(n^{\log_2 2})$$

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

$$a=2, b=2, k=2, p=0$$

$$\text{Hence, } a < b^k$$

$$\Rightarrow 2 < 2^2$$

$$\text{and } p \geq 0$$

$$\therefore T(n) = \Theta(n^2 \log^0 n)$$

$$= \Theta(n^2)$$

$$0 > n, n^2$$

$$0 \leq 3^2 (n \log 2)$$

$$(5) T(n) = 2T\left(\frac{n}{2}\right) + \frac{n^2}{\lg n}$$

$$a=2, b=2, k=2, f=-1$$

Here, $a < b$

$$s \leq s < \dots$$

$$\Rightarrow 2 \leq 2^2$$

$$L \rightarrow 2^k b^m$$

and $f < 0$

$$\therefore T(n) = \Theta(n^2)$$

$$(n^2) \Theta =$$

$$D) T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

$$a \geq 1, b \geq 1, f(n)$$

$$T(n) = n^{\log_b a} [U(n)]$$

$U(n)$ depends on $h(n)$

$$h(n) = \frac{f(n)}{n^{\log_b a}}$$

$f, h(n) \Theta = (n) U(n)$	
$n^n, n > 0$	$O(n^n)$
$n^n, n < 0$	$O(1)$
$(\log n)^c, c \geq 0$	$\frac{(\log_2 n)^{c+1}}{c+1}$

$$1 + \left(\frac{f}{b}\right) D = (n) D$$

$$b=2, a=2, s=d, L=n$$

$$f = 0 \text{ (constant)}$$

$$s = L \leftarrow$$

$$L < 2^k b^m$$

$$(n^2) \Theta =$$

$$\frac{n}{n^2} + \left(\frac{f}{b}\right) D \leq (n) D$$

$$b=2, a=2, s=d, L=n$$

$$f = 0 \text{ (constant)}$$

$$s = L \leftarrow$$

$$L < 2^k b^m$$

$$(n^2) \Theta = (n) D$$

$$(n^2) \Theta =$$

$$\begin{aligned}
 \text{Ex: (1)} \quad T(n) &= T\left(\frac{n}{2}\right) + 1 \\
 &= 1 \cdot T\left(\frac{n}{2} - 1\right) + 1 \\
 \text{hence } b &\neq 1
 \end{aligned}$$

Not solvable by Master method.

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

$$a=8, b=2, f(n)=n^2 \quad n + \left(\frac{n}{2}\right)P \stackrel{?}{=} (n)P \quad (1)$$

$$\begin{aligned}
 T(n) &= n^{\log_b a} [U(n)] \\
 &= n^{\log_2 8} [U(n)] \\
 &= n^3 U(n)
 \end{aligned}$$

$$h(n) = \frac{f(n)}{n^{\log_b a}} = \frac{n^2}{n^3} = \frac{1}{n} = n^{-1} \quad n + \left(\frac{n}{2}\right)P \stackrel{?}{=} (n)P \quad (2)$$

$$\therefore U(n) = O(1)$$

$$\text{Hence, } T(n) = n^3 \cdot O(1)$$

$$(3) T(n) = T\left(\frac{n}{2}\right) + c$$

$$\text{Hence, } a=1, b=2, f(n) = c \quad n + \left(\frac{n}{2}\right)P \stackrel{?}{=} (n)P \quad (3)$$

$$T(n) = n^{\log_b a} [U(n)]$$

$$= n^{\log_2 1} [U(n)] = U(n)$$

$$h(n) = \frac{f(n)}{n^{\log_b a}} = \frac{c}{n^0} = c = (\log_2 n)^0 \cdot c = c$$

$$U(n) = \frac{(\log_2 n)^{0+1}}{0+1}$$

$$L + \left(1 - \frac{N}{L}\right)P \cdot L =$$

2 KJ 22 N 20

$$= \log_2 n$$

$$\therefore T(n) = O(\log_2 n)$$

$$N + \left(\frac{N}{s}\right)P\delta = (N)P(s)$$

$$\text{④) (1) } T(n) = 2T\left(\frac{n}{2}\right) + n \quad \text{for } n = 1, 2, 3, \dots$$

a can't be function of n .

$$(2) T(n) = 64T\left(\frac{n}{2}\right) + n^2$$

$$(n) U^{\epsilon_N} =$$

$f(n)$ can't be negative. $\frac{1}{n} = \frac{1}{e^N} = \frac{e^{-N}}{e^N} = \frac{1}{e^{2N}}$ $= (N)^N$

$$(3) T(n) = T\left(\frac{n}{2}\right) + \sin n$$

$f(m)$ can't be \sin, \cos etc..

$$(4) T(n) = 2T\left(\frac{n}{2}\right) + \frac{1}{n}, \quad O(N) \leq T(n) \leq \Theta(N \log N)$$

k must be greater or equal zero ($k \geq 0$)

$$(5) T(n) = 0.5 T\left(\frac{n}{2}\right) + n$$

a must be greater or equal one ($a \geq 1$).

[(N)V] + N = (N)1

$$(\text{**})U = [(\text{**})U]^{\text{t}} \text{ for all } n =$$

Not solvable by
Master method.

$$(6) T(n) = \sqrt{2} T\left(\frac{n}{2}\right) + \log n$$

Hence, $a = \sqrt{2}$, $b = 2$, $k = 0$, $P = 1$

$$a > b^k$$

$$\Rightarrow \sqrt{2} > 2^0$$

$$\therefore T(n) = \Theta(n \log_b a)$$

$$= \Theta(n \log_2 \sqrt{2})$$

$$= \Theta(n^{1/2})$$

$$= \Theta(\sqrt{n})$$

$$10. \text{ Solve } T(n) = \begin{cases} T(\frac{n}{2}) + \log n & \text{if } n \geq 2 \\ O(1) & \text{else} \end{cases}$$

Sol:

$$\text{Let, } n = 2^m$$

$$n = 2^m \Rightarrow \log n = m \log_2 2 = m$$

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

$$\text{Let, } T(2^m) = S(m)$$

$$\therefore S(m) = S\left(\frac{m}{2}\right) + m$$

$$T(n) = a T\left(\frac{n}{b}\right) + n^k \log n$$

right = knight

$$T(n) = O(n) + m = (n)T(n)$$

Hence, $a = 1, b = 2, k = 1, P = 0$ $\therefore T(n) = O(n^2)$

$$a < b^k$$

$$\Rightarrow 1 < 2^1$$

and $P = 0$

$$\therefore S(n) = O(n^2 \log^0 n)$$

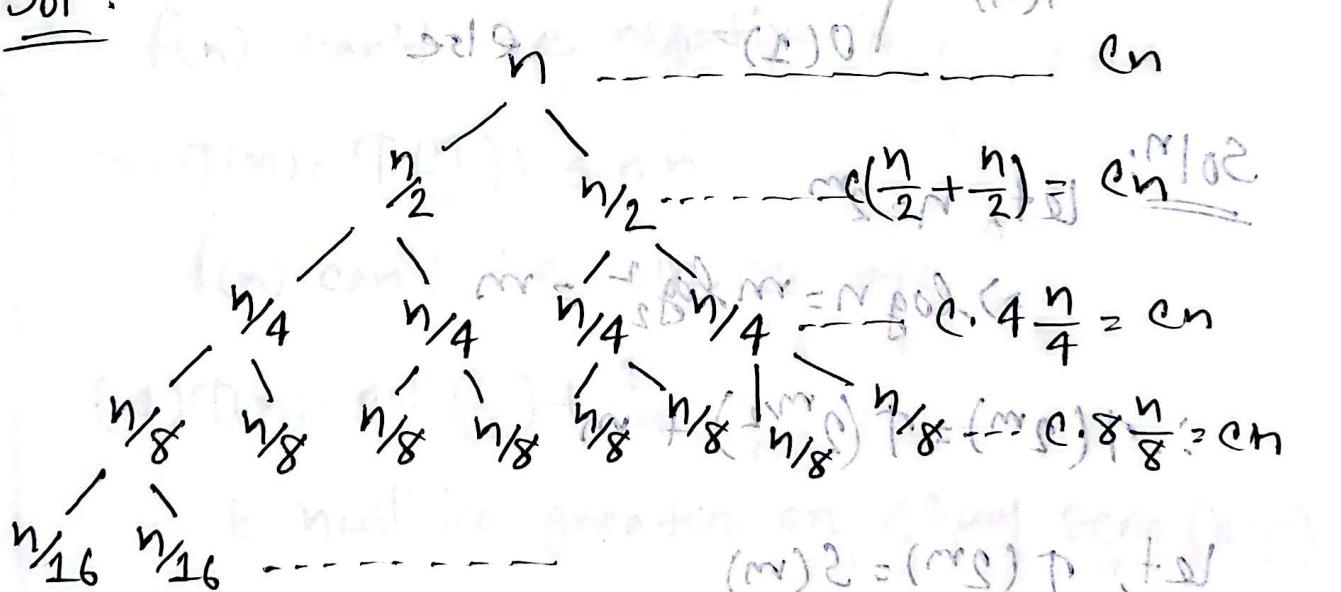
$$= O(n)$$

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

Q) Recursive Tree Method:

Q. Solve $T(n) = 2T\left(\frac{n}{2}\right) + cn$

Soln:

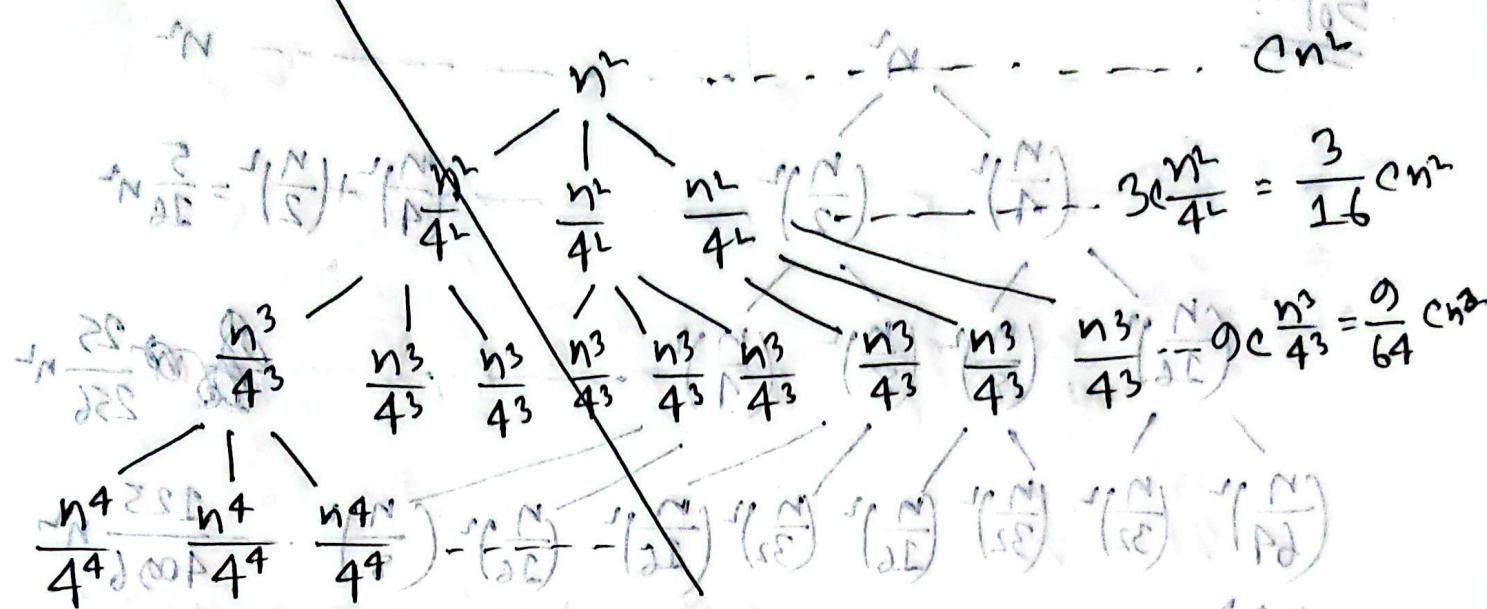


Total cost depends on the height.

Height = $\log n$

$$\therefore T(n) = cn * (\text{height}) = cn \log n \quad \therefore T(n) = O(n \log n)$$

Q. Solve $T(n) = 3T(\lfloor n/4 \rfloor) + cn^2$ (n)P = (n)P \Rightarrow n^2



Q. Solve $T(n) = 3T(\lfloor n/4 \rfloor) + cn^2$

Solⁿ: $\dots + n^2/400P + n^2/25P + n^2/16P + n^2 - (cn^2)$

$$\left[- \left(\frac{n}{4} \right)^2 \left(\frac{21}{2} \right) - \left(\frac{n}{4} \right)^2 \left(\frac{3}{2} \right) + 1 \right] 3c \left(\frac{n}{4} \right)^2 = \frac{3}{16} cn^2$$

$$\left(\frac{n}{16} \right)^2 \left(\frac{21}{2} \right) \left(\frac{n}{16} \right)^2 \left(\frac{3}{2} \right) \left(\frac{n}{16} \right)^2 \left(\frac{3}{2} \right) \dots 9c \left(\frac{n}{16} \right)^2 = \left(\frac{3}{16} \right)^2 cn^2$$

$$\left(\frac{n}{64} \right)^2 \left(\frac{n}{64} \right)^2 \left(\frac{n}{64} \right)^2 \dots \left(\frac{3}{16} \right)^2 \cdot n =$$

$$\therefore T(n) = cn^2 + \frac{3}{16} cn^2 + \left(\frac{3}{16} \right)^2 cn^2 + \dots \left(\frac{3}{16} \right)^n = (n)P$$

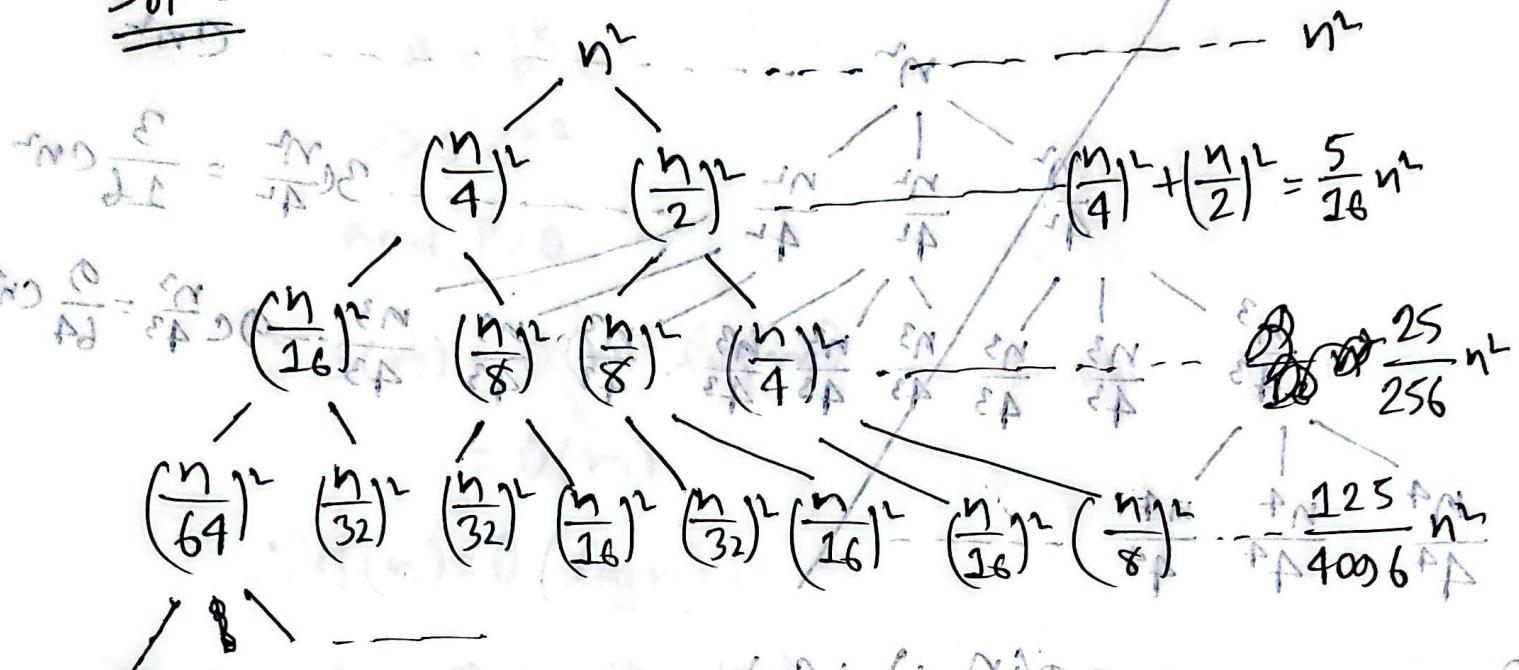
$$= cn^2 \left[1 + \frac{3}{16} + \left(\frac{3}{16} \right)^2 + \left(\frac{3}{16} \right)^3 + \dots \right]$$

$$= cn^2 \cdot \frac{1}{1 - \frac{3}{16}} = cn^2 \left(\frac{16}{13} \right)$$

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

Q. Solve $T(n) = T(n_1) + T(n_2) + n^2 \log n$ for n_1, n_2 .

Soln:



$$\therefore f(n) = n^2 + \frac{5}{16}n^2 + \frac{25}{256}n^2 + \frac{125}{4096}n^2 + \dots$$

$$\text{Ansatz: } \frac{d^2}{dt^2} = \left(\frac{5}{16}\right) n^2 \left[1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \left(\frac{5}{16}\right)^3 \left(\frac{t}{4}\right)^2 - \right]$$

$$= n^2 \cdot \left(\frac{16}{11} \right)$$

$$\therefore T(n) = O(n^2).$$

$$\therefore T(n) = O(n^2) \cdot \left(n^3 \left(\frac{c}{2L} \right) + M \frac{c}{2L} + M \right) = (n)^5 \therefore$$

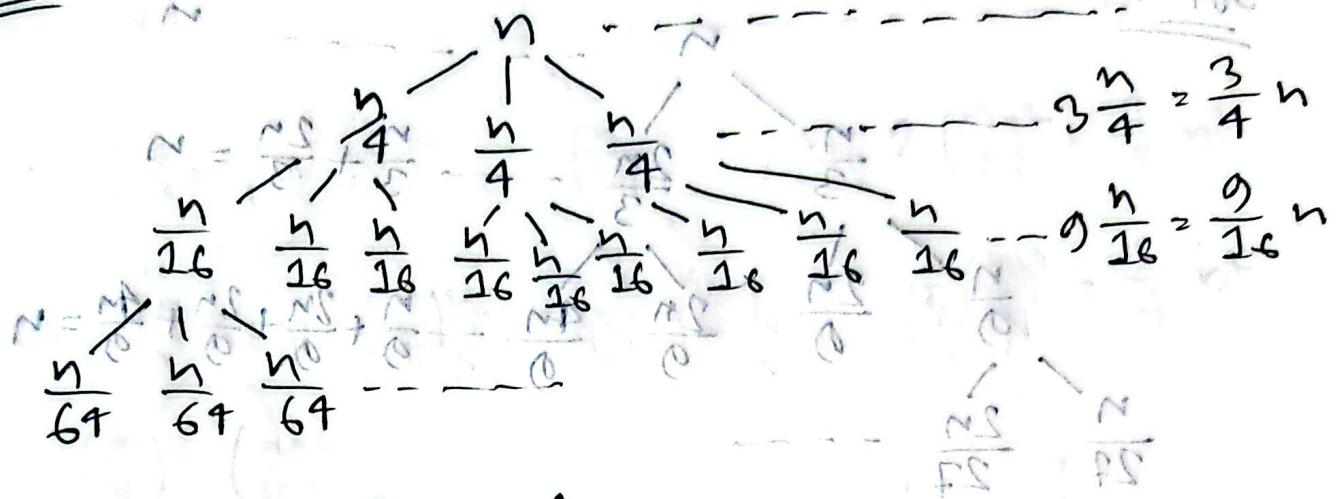
$$\left[-\frac{\partial \rho}{\partial \xi} + \left(\frac{\xi}{\partial \xi} \right) + \left(\frac{\xi}{\partial \xi} \right) + \frac{\xi}{\partial \xi} + 1 \right] \rho =$$

$$\left(\frac{dF}{dx}\right)_{x=0} = -\frac{F}{x_0 - x} \Big|_{x=0} =$$

$$(\mathbf{w})\mathbf{0} = (\mathbf{w})\mathbf{f}.$$

Q. Solve $T(n) = 3T\left(\frac{n}{4}\right) + n + \left(\frac{n}{4}\right)n = O(n^2)$ or $\Omega(n^2)$.

Sol:



$$\therefore T(n) = n + \frac{3}{4}n + \frac{9}{16}n + \dots$$

$$= n \left[1 + \frac{3}{4} + \left(\frac{3}{4}\right)^2 + \dots \right]$$

$$= n \cdot \frac{1}{1 - \frac{3}{4}}$$

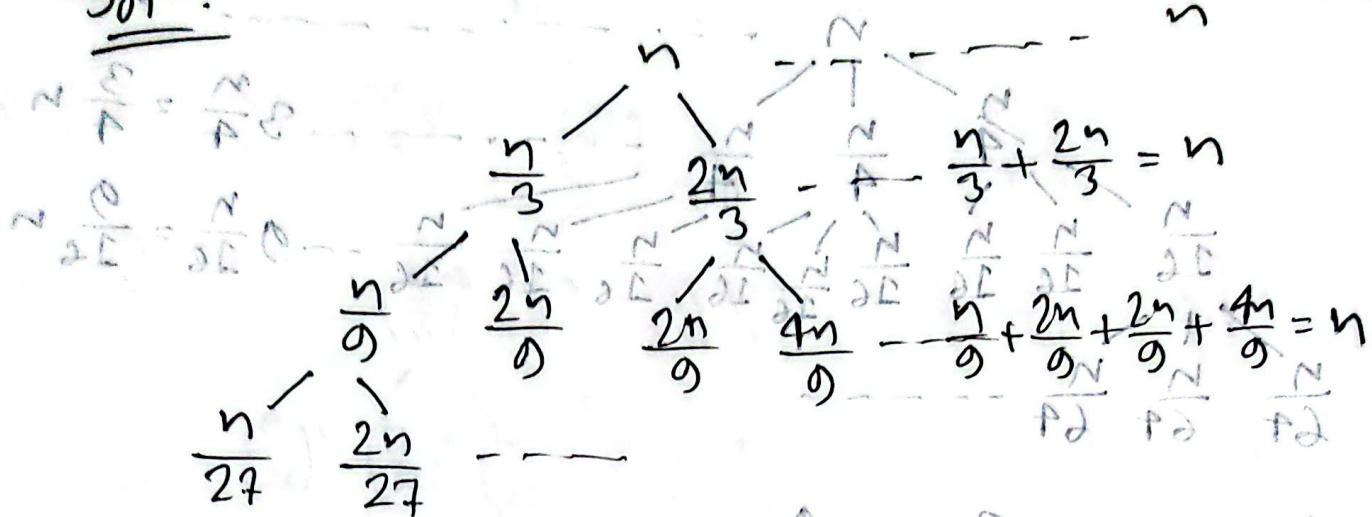
$$= n \cdot \frac{4}{1}$$

$$= 4n$$

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

Q. Solve $T(n) = T\left(\frac{n}{3}\right) + T\left(\frac{2n}{3}\right) + n$: M102

Soln:



Total cost depends on height

$$\text{Height} = \log_{3/2} n \approx \log n$$

$$\therefore T(n) = n * \text{height}$$

$$= n \log n$$

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

$$\frac{1}{\frac{3}{2} - 1} \cdot n =$$

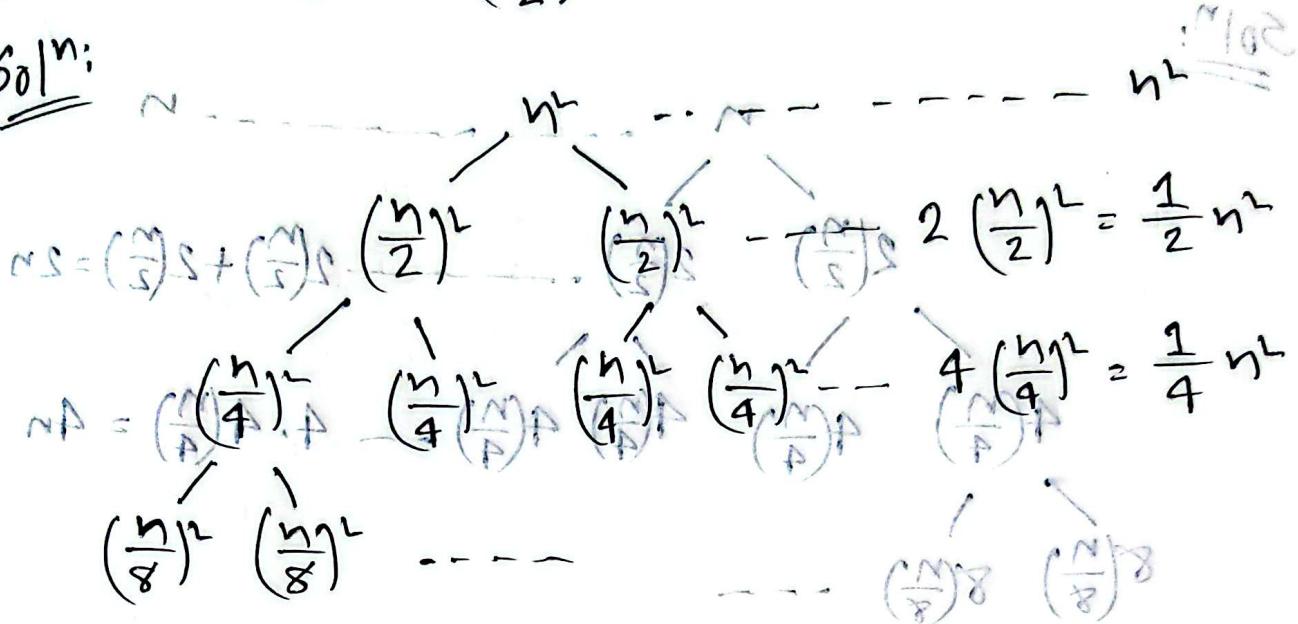
$$\frac{2}{\frac{1}{2}} \cdot n =$$

$$n^2 =$$

$$, (n)O = (n)P. \therefore$$

$$Q. \text{ Solve } T(n) = 2T\left(\frac{n}{2}\right) + n^2 \quad (n^2 \text{ is } O(n^2)) \quad \text{Ans: } O(n^2)$$

Sol:



$$\therefore T(n) = n^2 + \frac{1}{2}n^2 + \frac{1}{4}n^2 + \dots + n^2 + n^2 + \dots = n^2 + n^2 + \dots = n^2$$

$$= n^2 \left[1 + \frac{1}{2} + \frac{1}{4} + \dots \right] = n^2 \left[\frac{1}{1 - \frac{1}{2}} \right] =$$

$$= n^2 \cdot \frac{1}{1 - \frac{1}{2}} =$$

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

$$= 2n^2$$

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

$$= \frac{n^2}{1 - \frac{1}{2}} \cdot n^2 =$$

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

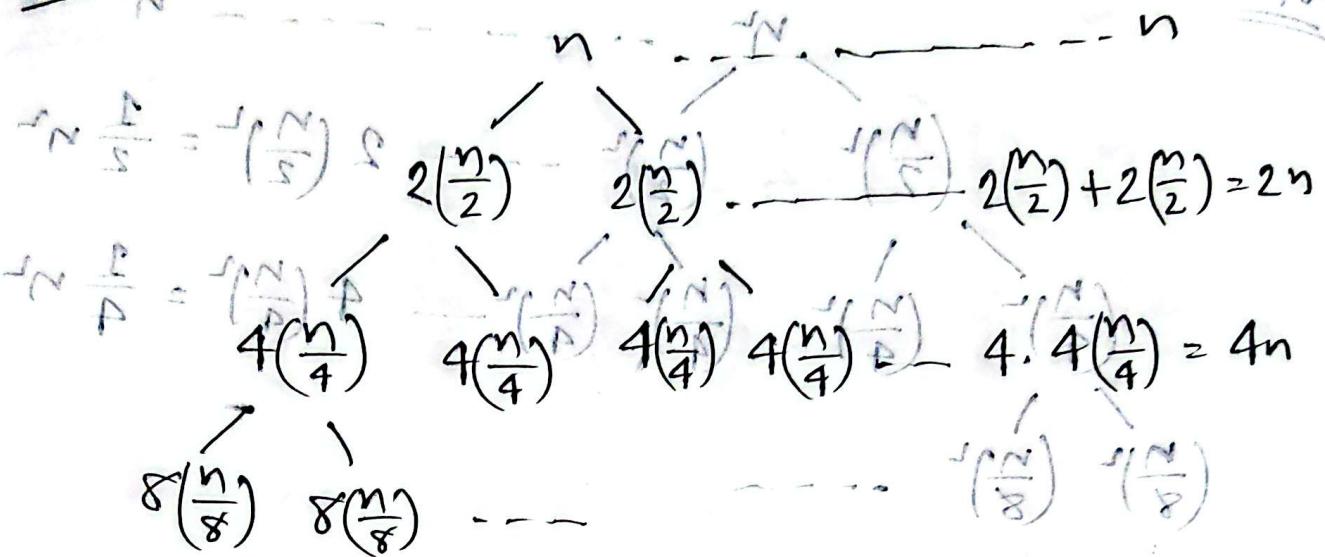
$$= \frac{2n^2}{1} \cdot n^2 =$$

$$= 2n^4$$

$$\therefore O(n^4) = O(n^2) \quad \text{Ans: } O(n^2)$$

Q. Solve $T(n) = 4T\left(\frac{n}{2}\right) + n$ (Recurrence Relation)

Sol:



$$\therefore T(n) = n + 2n + 4n + \dots + n \log n + n = (n)T$$

$$= n \left[1 + 2 + 4 + \dots + \log n \right] + n =$$

$$= n \cdot \frac{1 - 2^{\log n}}{1 - 2}$$

$$= n \cdot \frac{1 - n}{1 - 2}$$

$$= n \cdot \frac{n-1}{2-1}$$

$$= n^2 - n$$

$$\frac{1}{2} - \frac{1}{2}$$

$$\frac{2}{2} \cdot n =$$

$$n^2 - n$$

$$(n)O = (n)T$$

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