$an) = n \log(n)$

① i) Show that for any real constants a and b, where $b>0$, $(n+a)^b = O(n^b)$.

Ans:- Here $f(n) = (n+a)^b$

$g(n) = n^b$

using limit notation

$$C = \lim_{n\to\infty} \frac{f(n)}{g(n)}$$

$$= \lim_{n\to\infty} \frac{(n+a)^b}{n^b}$$

$$= \lim_{n\to\infty} \left(\frac{\cancel{n}(1+\frac{a}{n})}{\cancel{n}}\right)^b$$

$$= \lim_{n\to\infty} \left(1+\frac{a}{n}\right)^b$$

$$= \lim_{n\to\infty} 1^b + \cancel{\lim_{n\to\infty} \frac{a}{n}} \rightarrow 0$$

$$= \lim_{n\to\infty} 1^b$$

$$= 1$$

So $\lim_{n\to\infty} \frac{(n+a)^b}{n^b} = 1 < \infty \Rightarrow (n+a)^b = \Theta(n^b)$

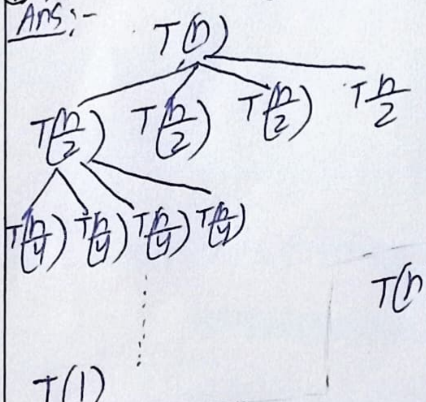
As we know that Θ can be converted to O

so, $(n+a)^b = O(n^b)$ for all real a and $b>0$

Hence proved.

ⅱ) solve $T(n) = 4T(\frac{n}{2}) + n^2$.

Ans:-



cost

n^2

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

$16(\frac{n}{4})^2 = n^2$

$T(n) = n + \frac{n}{2} + \frac{n}{2^2} + \dots + \frac{n}{2^k}$

$= n + \frac{n}{2} + \frac{n}{2^2} + \dots + 1$ $\{\because$ Let $\frac{n}{2^k} = 1\}$

$\Rightarrow \log n = k \log 2$

$\Rightarrow k = \log n$

Total cost $= n^2 + n^2 + n^2 + \dots + k$

$= n^2 + n^2 + \dots \log n$

$= O(n^2 \log n)$

$\therefore$ The total time complexity is $O(n^2 \log n)$

$f(n) = n \log(n)$

(iii) Show that $f1(n) + f2(n) = O(\max(g1(n), g2(n)))$

Ans:- Here $f1(n) = O(g_1(n))$ $\quad f2(n) = O(g_2(n))$

$\Rightarrow f1(n) \leq c_1 g_1(n)$ $\quad \Rightarrow f2(n) \leq c_2 g_2(n)$

Now adding both the function, we get

$f1(n) + f2(n) \leq c_1 g_1(n) + c_2 g_2(n)$

$f1(n) + f2(n) \leq (c_1 + c_2) \max(g_1(n), g_2(n))$ {∵ using maximum method}

$\therefore f1(n) + f2(n) = O(\max(g_1(n), g_2(n)))$

Hence proved

2) (i) Solve $T(n) = 3T(n^{1/3}) + \log^3 n$.

Ans:-

let $n = 2^m$

$\Rightarrow m = \log n$

$T(2^m) = 3T(2^{m/3}) + m^3$

let $T(2^m) = S(m)$

$S(m) = 3S(m/3) + m^3$

using master's method

$a = 3$ $\quad b = 3$ $\quad f(m) = m^3$

$m^{\log_b a} = m^{\log_3 3} = m^1$

$m^3 > m^1$

$\Rightarrow f(m) > m^{\log_b a}$ (∵ case-3 applied)

$\Rightarrow a \cdot f\left(\frac{2m}{3}\right) \leq c\,2^m$

$\Rightarrow \frac{1}{3} \leq c$

$\Rightarrow c = \frac{1}{3}$

$S(m) = \Theta(m)$

$\Rightarrow S(m) = \Theta(m^3)$

$\Rightarrow T(n) = \Theta(\log^3 n)$

(∴ Ans)

② ii) solve $T(n) = 3T(n/4) + n\log n$. (using master theorem).

<u>Ans</u>:- Here $a=3$ $b=4$ $f(n) = n\log(n)$

$n^{\log_b a} = n^{\log_4 3} \approx n^{0.79}$

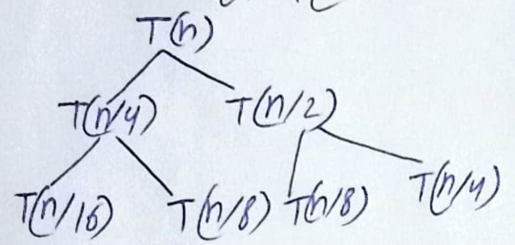
$n\log n > n^{0.79}$

$\Rightarrow f(n) > n^{\log_b a}$ {∴ case-3 applied}

& $f(n) = \Omega(n^{0.79+\epsilon})$

$\Rightarrow T(n) = \Theta(n\log n)$ (Ans)

② iii) solve $T(n) = T(n/4) + T(n/2) + cn^2$ (using Recurrence Tree).



<u>cost</u>

cn^2

$cn^2 \frac{5}{16}$

$cn^2 \left(\frac{5}{16}\right)^2$

$T(n) = T(n/4) + T(n/2) + cn^2$

cost of each level is cn^2

so for $T(n/4) \Rightarrow c\left(\frac{n^2}{4}\right) = \frac{cn^2}{16}$

so for $T(n/2) \Rightarrow c\left(\frac{n^2}{2}\right) = \frac{cn^2}{4}$

Total cost at level 1 $= \frac{cn^2}{16} + \frac{cn^2}{4}$

$= cn^2\left(\frac{1}{16} + \frac{1}{4}\right)$

$= cn^2 \cdot \frac{5}{16}$

so costs of level 0 : cn^2

level 1 : $cn^2 \cdot \frac{5}{16}$

level 2 : $cn^2 \left(\frac{5}{16}\right)^2$

level 3 : $cn^2 \cdot \left(\frac{5}{16}\right)^3$

$T(n) = cn^2\left[1 + \frac{5}{16} + \left(\frac{5}{16}\right)^2 + \cdots\right]$

$= cn^2\left(\frac{1}{1-\frac{5}{16}}\right)$

$= cn^2 \frac{16}{11}$

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