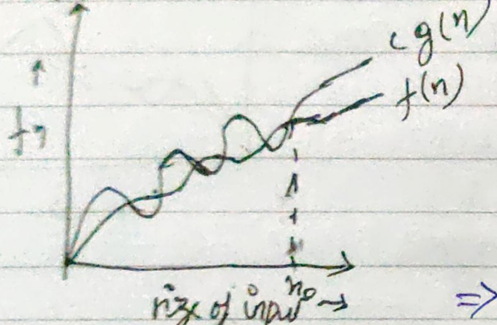
Ques. Asymptotic Notation :-

→ Tending to infinity

They help us find the complexity of an algorithm when input is very large

1) Big O(o)



$f(n) = O(g(n))$

iff $f(n) \leq c.g(n)$

$\forall n \geq n_0$

for some constant $c > 0$

$\Rightarrow g(n)$ is tight upper bond of $f(n)$

2) Big Omega (Ω)

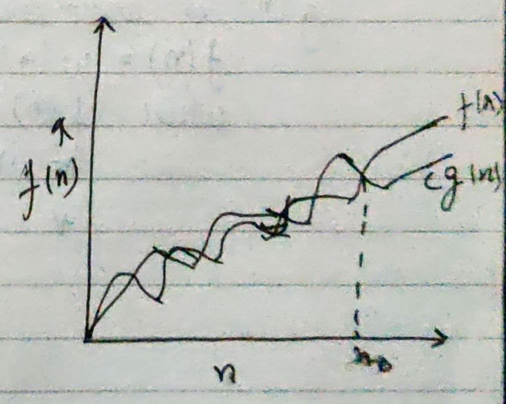
$f(n) = \Omega(g(n))$

$g(n)$ is tight lower bond of $f(n)$

$f(n) = \Omega(g(n))$

iff $f(n) \geq c\,g(n)$

$\forall n \geq n_0$ for some $c > 0$



3 Theta (θ)

$f(n) = \theta(g(n))$

$g(n)$ is both tight upper & lower bound of fn $f(n)$

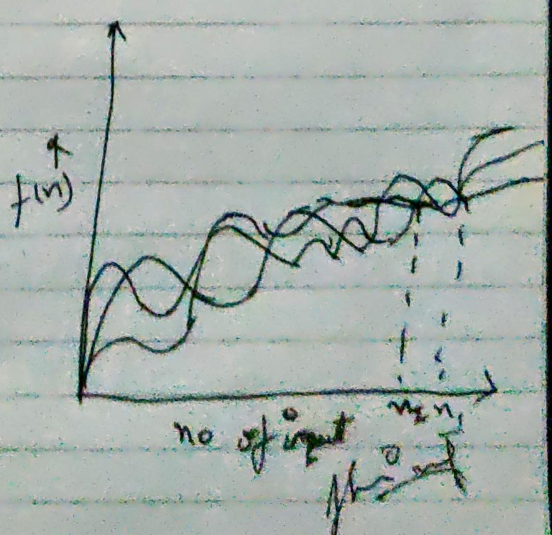
$f(n) = \theta(g(n))$

iff

$c_1 g(n) \leq f(n) \leq c_2 g(n)$

$\forall n \geq \max(n_1, n_2)$

for some constant $c_1 > 0$ & $c_2 > 0$



4) small o(o)

$f(n) = o(g(n))$

$g(n)$ is upper bound of fn $f(n)$

$f(n) = o(g(n))$

when $f(n) < c.g(n)$

$\forall\ n > n_0$

& $\forall\ c > 0$

5) small omega (ω)

$f(n) = \omega(g(n))$

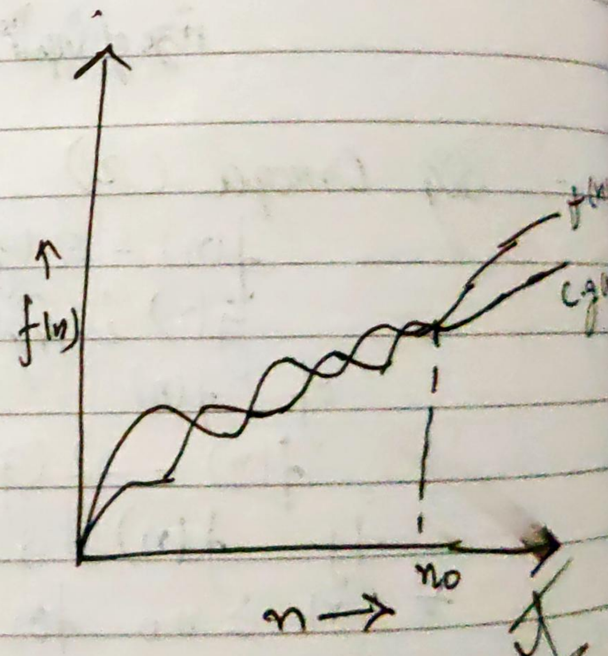
$g(n)$ is lower bound of fn $f(n)$

$f(n) = \omega(g(n))$

when $f(n) > c.g(n)$

$\forall\ n > n_0$

$\forall\ c > 0$



Que 2) for (i = 1 to n) // i = 1, 2, 4, 8 - - - n
{ i = i * 2) // O(1)

$$\Rightarrow \sum_{i=1}^{n} 1 + 2 + 4 + 8 + \cdots n$$

G.P k^{th} value $\Rightarrow T_K = ar^{K-1}$

$$= 1 \times 2^{K-1}$$

$$n = 2^{K-1}$$

$$2n = 2^K$$

$$\log n = \log 2^K$$

$$\log 2 + \log n = K \log 2$$

$$\log_2 n + 1 = K$$

$$\Rightarrow O(K) = O(1 + \log_2 n)$$

$$= O(\log n)$$

Ques 3) $T(n) = 3T(n-1)$ — ①

put $n = n-1$ in ①

$T(n-1) = 3T(n-2)$ — ②

from 1 & 2

$T(n) = 9T(n-2)$ — ③

put $n = n-2$ in ①

$T(n-2) = 3T(n-3)$ — ④

from 1 & 4

$T(n) = 27(T(n-3))$ — 5

from this we get

$T(n) = 3^K T(n-K)$.

put $n - K = 0$

$n = K$

$T(n) = 3^n T(n-n)$

$T(n) = 3^n T(0)$

$T(n) = 3^n \times 1$ $\quad [\because T(0) = 1]$

$\boxed{\therefore T(n) = O(3^n)}$

Ques 4) $T(n) = 2T(n-1) - 1$ — ①

put $n = n-1$

$T(n-1) = 2T(n-2) - 1$ — ②

put 2 in 1

$T(n) = 4T(n-2) - 2 - 1$ — ③

put $n = n-2$

$T(n-2) = 2T(n-3) - 1$ — ④

put 4 in 3

$T(n) = 8T(n-3) - 4 - 2 - 1$ — ⑤

from 5 we get

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

$G.P = 2^{k-1} + 2^{k-2} + \cdots 1$

$a = 2^{k-1} \quad r = 1/2$

sum of $G.P = \frac{a(1-r^n)}{1-r} = 2^{k-1}\frac{(1-(1/2)^k)}{1-1/2}$

$= 2^k(1-(1/2)^k)$

$= 2^k - 1$

let $n-k = 0$

$n = k$

$$T(n) = 2^n[T(n-n)] - (2^k - 1)$$
$$= 2^n \cdot 1 - 2^n + 1$$
$$T(n) = 2^n - 2^n + 1 \qquad (\because T(0) = 1)$$

$$\boxed{\therefore T(n) = O(1)}$$

Ques 5)

$i = 1 \quad 2 \quad 3 \quad 4 \quad 5 \quad 6 \quad \dots$
$S = 1 + 3 + 6 + 10 + 15 + 21 \dots n$

sum of $S = 1 + 3 + 6 + 10 + \dots + n$ — ①
also $S = 1 + 3 + 6 + 10 + \dots T_{n-1} + T_n$ — ②

from 1 − 2

$$0 = 1 + 2 + 3 + 4 + \dots n - T_n$$

$$T_k = 1 + 2 + 3 + 4 + \dots k$$
$$T_k = \frac{1}{2} k(k+1)$$

for k iterations

$$1 + 2 + 3 + \dots + k \le n$$
$$\frac{k(k+1)}{2} \le n$$
$$n \ge \frac{k^2 + k}{2}$$

$$O(k^2) \le n$$
$$k = O(\sqrt{n})$$
$$\boxed{T(n) = O(\sqrt{n})}$$

Ques 6) as $i^2 \le n$

$i \le \sqrt{n}$

$i = 1, 2, 3, 4 \cdots, \sqrt{n}$

$\sum_{i=1}^{n} 1 + 2 + 3 + \cdots + \sqrt{n}$

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

$T(n) = \frac{n + \sqrt{n}}{2}$

$\boxed{T(n) = O(n)}$

Ques 7) for $K = K^2$

$K = 1, 2, 4, 8, \cdots n$

GP $\Rightarrow$ $a = 1$ $r = 2$

Sum $= \frac{1(r^n - 1)}{r-1}$

$= \frac{1(2^k - 1)}{1}$

$n = 2^k - 1$

$K = \log n$

i	j	K
1	$\log n$	$\log n \times \log n$
2	$\log n$	$\log n \times \log n$
⋮	⋮	⋮
n	$\log n$	$\log n \times \log n$

$O(n * \log n * \log n)$

$\boxed{O(n \log^2 n)}$

Ans 8)

$i = 1, 2, 3, 4 \ldots n \Rightarrow O(n)$
$j = 1, 2, 3, 4 \ldots n^2 \Rightarrow O(n^2)$

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

$$a = 1, \quad b = 3 \quad f(n) = n^2$$

$$c = \log_3 1 = 0$$

$$n^0 = 1 \qquad f(n) = n^2$$

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

Ans 9)

for $i = 1 \Rightarrow j = 1, 2, 3, 4 \ldots n = n$
for $i = 2 \Rightarrow j = 1, 3, 5 \ldots n = n/2$
for $i = 3 \Rightarrow j = 1, 4, 7 \ldots n = n/3$

for $i = n \Rightarrow j = 1 \qquad n$

$$\Rightarrow \sum_j = n + n/2 + n/3 + n/4 + \ldots 1$$

$$\sum_j = n\left[1 + 1/2 + 1/3 + 1/4 + \ldots + 1/n\right]$$

$$\sum_j = n[\log[n]]$$

0

$\boxed{T(n) = [n \log n]}$

Ques 10) as given n^k & c^n

relation b/w n^k & c^n is

$$n^k = O(c^n)$$

as

$$n^k \leq a\, c^n$$

$\forall\ n \geq n_0$ & some constant $a > 0$

for $n_0 = 1$

$c = 2$

$\Rightarrow 1^k \leq a\, 2^1$

$$\boxed{n_0 = 1 \ \& \ c = 2}$$