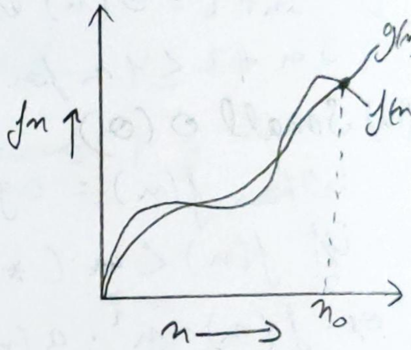
Tutorial - 1

Name - Harsh Agarwal
Section - F
Roll no. - 59
Univ. Roll no. - 2016757

Q1: What do you understand by Asymptotic notation, define diff. asymptotic notation with example.

(i) Big O(n)



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

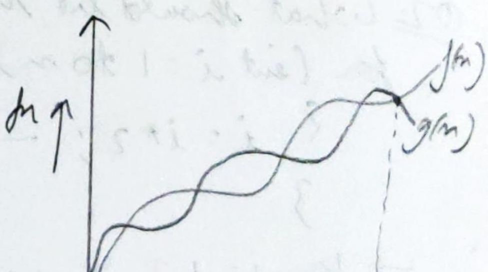
if $f(n) \leq g(n) \times C \quad \forall n \geq n_0$
for some constant, $C > 0$.
g(n) is 'tight' upper bound of f(n).

eg → $f(n) = n^2 + n$
$g(n) = n^3$
$n^2 + n \leq C * n^3$
$n^2 + n = O(n^3)$

(ii) Big Omega (Ω)

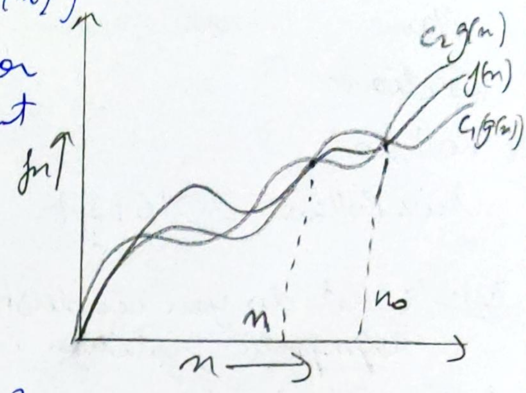
When $f(n) = \Omega(g(n))$ means $(g(n))$ is "tight" lowerbound of f(n) i.e. f(n) can go beyond g(n) i.e. $f(n) = \Omega\, g(n)$.

iff $f(n) \geq C \cdot g(n)$
$\forall n_2 > n_0$ & C = constant > 0



Eg: $f(n) = n^3 + 4n^2$
$g(n) = n^2$
i.e. $f(n) \geq C * g(n)$
$n^3 + 4n^2 = \Omega(n^2)$

iii) Big Theta (Θ) - When $f(n) = \Theta(g(n))$ gives the tight upperbound & lowerbound both. i.e. $f(n) = \Theta(g(n))$ iff $c_1 * g(n_1) \leq f(n) \leq c_2 * g(n_2)$ for all $n \geq \max(n_1, n_2)$, some constant ~~$c_1 * g(n$~~ $c_1 > 0$ & $c_2 > 0$. i.e. $f(n)$ can never go beyond $c_2\, g(n)$ & will never come down of $c_1\, g(n)$.



eg- $3n+2 = \Theta(n)$ as $3n+2 \geq 3n$ &

$3n+2 \leq 4n$ for n, $c_1 = 3$, $c_2 = 4$ & $n_0 = 2$.

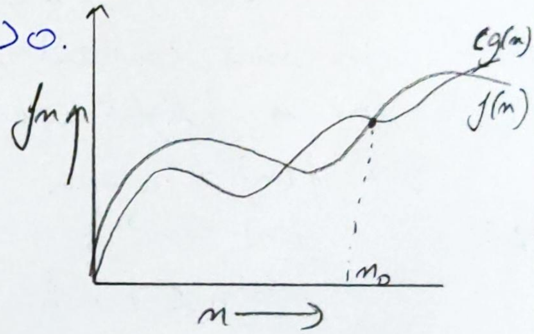
(iv) Small o (o)

When $f(n) = o\,g(n)$ gives the upper bound if $f(n) = o\,g(n)$ iff $f(n) < c * g(n)\ \forall n > n_0$ & $n > 0$.

ex- $f(n) = n^2$; $g(n) = n^3$

$f(n) < c * g(n)$

$n^2 = o(n^3)$

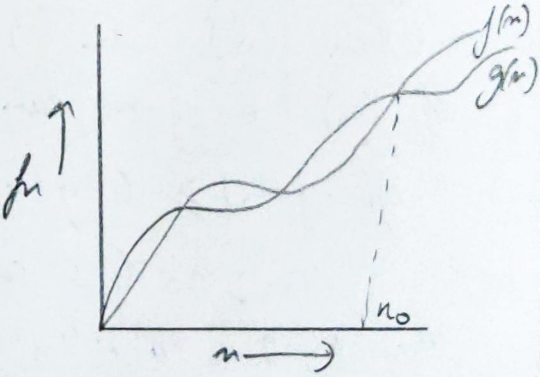


(v) Small Omega (ω)

It gives the 'lower bound' i.e. $f(n) = \omega(g(n))$.

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

iff $f(n) > c * g(n)\ \forall n > n_0$ & $c > 0$.



Q2: What should be time complexity of:

```
for (int i = 1 to n)
    {   i = i*2;   -> O(1)
    }
```

$\rightarrow$ for $i = 1, 2, 4, \ldots n$ times

i.e. series is a G.P.

so, $a = 1$, $r = 2$.

K^{th} value of GP:

$$t_k = ar^{k-1}$$
$$t_k = 1(2)^{k-1}$$
$$2n = 2^k$$
$$\log_2(2n) = k\log 2$$
$$\log_2 2 + \log_2 n = k$$
$$\log_2 n + 1 = k$$

So, time complexity $T(n) \Rightarrow O(\log_2 n)$

Q3. $T(n) = \{ 3T(n-1)$ if $n > 0$ otherwise $1 \}$

↳ i.e. $T(n) \Rightarrow 3T(n-1)$ — ①

$$T(n) = 1$$

put $n = n-1$ in ①

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

put ② in ①

$$T(n) = 3(3T(n-2))$$
$$T(n) = 9T(n-2) \quad \text{— ③}$$

put $n = n-2$ in ①

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

put in ③

$$T(n) = 27T(n-3) \quad \text{— ④}$$

So, $T(k) = 3^k \, T(n-k)$ — 5

for k^{th} term, Let $n-k=1$ (base case)

$k = n-1$, put in 5

$T(n) = 3^{n-1} \, T(1)$

$T(n) = 3^{n-1}$

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

Q4. $T(n) = \{2T(n-1) - 1$ if $n > 0$, otherwise $1\}$

$T(n) = 2T(n-1) - 1$ — 1

put $n = n-1$

$T(n-1) = 2T(n-2) - 1$ — 2

put in 1

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

$= 4T(n-2) - 2 - 1$ — 3

put $n = n-2$ in 1

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

put in 1

$T(n) = 8T(n-3) - 4 - 2 - 1$ — 4

So, $T(n) = 2^k \, T(n-k) - 2^{k-1} - 2^{k-2} - \dots 2^0$.

⇒ k^{th} term

Let $n-k=1$

$k = n-1$

$T(n) = 2^{n-1} \, T(1) - 2^k \left(\frac{1}{2} + \frac{1}{2^2} + \dots + \frac{1}{2^n}\right)$

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

∴ a G.P.

$a = \frac{1}{2}$, $r = \frac{1}{2}$

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

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

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

$T(n) = \underline{O(1)}$

Q5.

```
int i=1, s=1;
while (s<=n)
  {
    i++;
    s = s+i;
    printf("#");
  }
```

$\hookrightarrow$ $i = 1 \quad 2 \quad 3 \quad 4 \quad 5 \ \ldots\ldots$

$S = 1 + 3 + 6 + 10 + 15 \ldots\ldots$

Sum of $S = 1 + 3 + 6 + 10 + \ldots\ldots n$ — (1)

Also, $S = 1 + 3 + 6 + 10 + \ldots T_{n-1} + T_n$ — (2)

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

$T_k = 1 + 2 + 3 + 4 + \ldots\ldots k$

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

for k, $1 + 2 + 3 + \ldots k <= n$

$\frac{k(k+1)}{2} <= n$

$\frac{k^2 + k}{2} <= n$

$O(k^2) \leq n$

$k = O(\sqrt{n})$

$T(n) = \underline{O(\sqrt{n})}$.

Q.6:

```
void f(int n)
{ int i, count = 0;
  for(int i = 1; i * i <= n ; ++i)
}
```

$\hookrightarrow$ As $i^2 = n$

$i = \sqrt{n}$

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

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

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

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

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

Q7.

```
void f(int n)
{
   int i, j, k, count = 0;
   for (int i = n/2 ; i<=n ; j = j*2)
      for (k=1 ; k <=n; k=k+2)
          count++;
}
```

$\hookrightarrow$ Since, for $n = k^2$

$k = 1, 2, 4, 8 \dots n$

$\because$ Series is in G.P.

So, $a=1$, $r=2$

$$\frac{a(r^k-1)}{r-1}$$

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

$$n = 2^k - 1$$

$$n+1 = 2^k$$

$\log_2(n) = k$

i	j	k
1	$\log(n)$	$\log(n) * \log(n)$
2	$\log(n)$	$\log(n) * \log(n)$
⋮	⋮	⋮
n	$\log(n)$	$\log(n) * \log(n)$

T.C $\Rightarrow O(n * \log n * \log n)$

$\Rightarrow \underline{O(n \log^2(n))}$

Q8.

```
void function(int n)
{   if (n==1) return;
    for(i=1 to n){
     for(j=1 to n){
       printf("*");
     }
  } function(n-3);
}
```

↳ for (i = 1 to n)

we get j = n times every time

$\therefore \; i * j = n^3$

k^{4}, Now, $T(n) = n^2 + T(n-3);$

$T(n-3) = (n^2-3)^2 + T(n-6);$

$T(n-6) = (n^2-6)^2 + T(n-9);$

and $T(1) = 1;$

Now, put these values in $T(n)$.

$T(n) = n^2 + (n-3)^2 + (n-6)^2 + \cdots + 1.$

Let $n - 3k = 1$

$k = (n-1)/3$ total terms $= k+1$

$T(n) = n^2 + (n-3)^2 + (n-6)^2 + \cdots + 1.$

$T(n) \simeq k n^2$

$T(n) \simeq (k-1)/3 * n^2$

So, $T(n) = \underline{O(n^3)}$

Q9.

```
void f(int n){
 for(int i=1 to n){
 for(int j=1; j<=n; j=j+1){
   printf("*");
 }}
}
```

→ for $i=1$ $\quad j = 1+2+\cdots(n>, j+i)$

$i=2$ $\quad j = 1+3+5+\cdots(n>, j+i)$

$i=3$ $\quad j = 1+4+7+\cdots(n \geq j+i)$

n^{th} term of AP is

$T(n) = a + d * m$

$T(m) = 1 + d * m$

$(n-1)/d = m$

for $i=1$ $\quad (n-1)/1$ times

$i=2$ $\quad (n-1)/2$ times

$i=n-1$

we get,

$T(n) = i_1 j_1 + i_2 j_2 + \cdots i_{n-1} j_{n-1}$

$= \frac{(n-1)}{2} + \frac{(n-2)}{2} + \cdots \cdot 1$

$= n + \frac{n}{2} + \frac{n}{3} + \cdots n/n-1 - n \times 1$

$= n[1 + 1/2 + 1/3 + \cdots 1/n-1] - n \times 1$

$= n \times \log n - n + 1$

Since $\int 1/x = \log x$

$T(n) = \underline{O(n \log n)}$

Q10. For the function n^k & C^n, what is the asymptotic Relationship b/w these functions? Assume that $k >= 1$ & $C > 1$ are constants. Find out the value of C & no. of which relationships holds.

↳ As given n^k & C^n

Relationship b/w n^k & C^n is

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

$n^k \leq a(C^n)$

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

for $n_0 = 1$; $C = 2$

$\Rightarrow 1^k = a^2$

$\Rightarrow \underline{n_0 = 1 \text{ \& } C = 2}$