

Q1- What do you understand by Asymptotic notations.
Define different Asymptotic Notation with examples.

\Rightarrow (i) Big O(n)

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

if $f(n) \leq g(n) \times c \forall n > n_0$

for some constant, $c > 0$

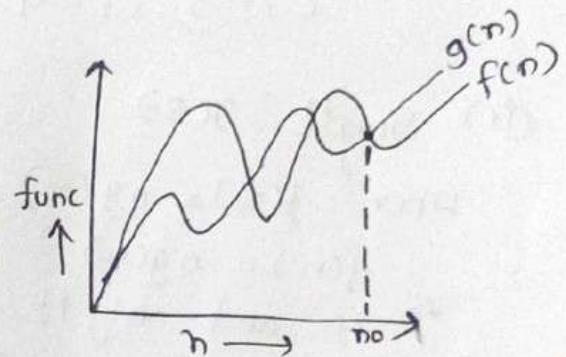
$g(n)$ is 'tight' upper bound of $f(n)$

$$\text{Ex} \rightarrow f(n) = n^2 + n$$

$$g(n) = n^3$$

$$n^2 + n \leq c * n^3$$

$$n^2 + n = O(n^3)$$



(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 $g(n)$

$$\text{i.e. } f(x) = \Omega g(x)$$

if and only if

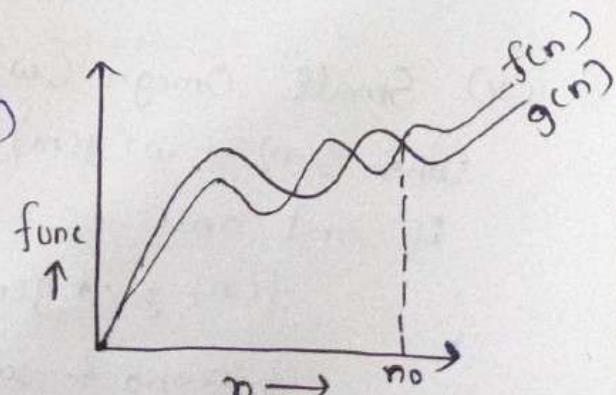
$$f(n) \geq c \cdot g(n)$$

$$\text{Ex} \rightarrow f(n) \Rightarrow n^3 + 4n^2$$

$$g(n) = n^2$$

$$\text{i.e. } f(n) \geq c \cdot g(n)$$

$$n^3 + 4n^2 = \Omega(n^2)$$



(iii) Big Theta (Θ)

When $f(n) = \Theta(g(n))$

It gives tight upperbound and lowerbound both.

i.e. $f(n) = \Theta(g(n))$

if and only if

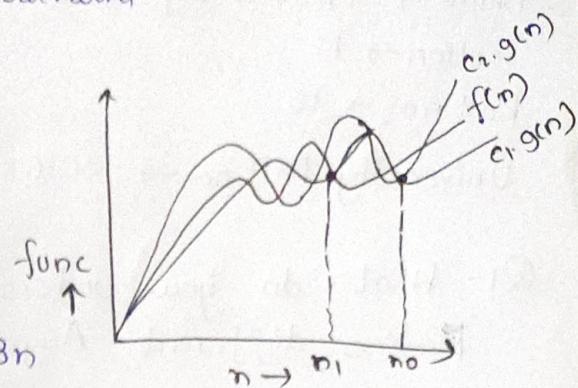
$$c_1 * g(n) \leq f(n) \leq c_2 * g(n)$$

$\forall n > \max(n_1, n_2)$, some constant $c_1, c_2 > 0$.

Ex $\rightarrow 3n+2 = \Theta(n)$ as $3n+2 \geq 3n$

& $3n+2 \leq 4n$ for n

$$c_1 = 3, c_2 = 4 \text{ & } n_0 = 2$$

(iv) Small O(O)

When $f(n) = o(g(n))$ gives the upper bound

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

if and only if

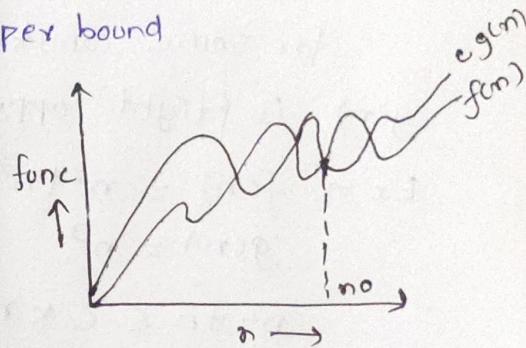
$$f(n) < c g(n)$$

$\forall n > n_0 \text{ & } n > 0$

Ex $\rightarrow f(n) = n^2 ; g(n) = n^3$

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

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

(v) Small Omega (ω)

When $f(n) = \omega(g(n))$ gives the lower bound

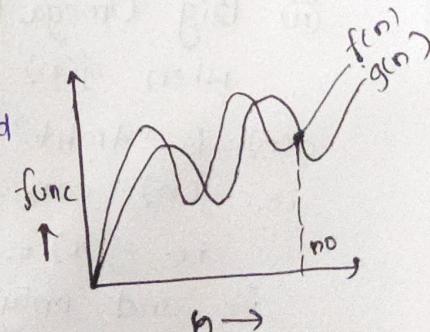
if and only if

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

$\forall n > n_0 \text{ & some constant } c > 0$

For ex $\rightarrow f(n) = n^2 \quad g(n) = n$

$$\therefore n^2 = \omega(n)$$



Q2 - What should be time complexity of
 $\text{for } (i=1 \text{ to } n)$
 $\quad \quad \quad t = i^2 L;$
 $\}$

$\Rightarrow i = 1, 2, 4, 6, 8, 10, \dots, n \text{ times}$

It is an G.P.

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

To find the k^{th} term

$$t_k = a \cdot r^{k-1}$$

$$n = (1) (2)^{k-1}$$

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

$$2n = 2^k$$

app. \log_2 on both sides

$$\log_2(2n) = \log_2(2^k)$$

$$\log_2 2 + \log_2 n = k \log_2 2$$

$$1 + \log_2 n = k$$

$$k = \log_2 n$$

T.C. $\Rightarrow O(\log_2 n)$

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

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

$$T(1) = 1$$

Put $n = n-1$ in (1)

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

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

put the value in (1)

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

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

Put $n = n-2$ in ①

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

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

Put the value in ②

$$T(n) = 9[3T(n-3)]$$

$$T(n) = 27T(n-3)$$

Generalizing Series

$$T(n) = 3^k T(n-k) \quad \text{--- ③}$$

for k^{th} terms, Let $n-k=1$

$$k=n-1$$

put in ③

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

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

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

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

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

Put $n = n-1$

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

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

Put in ①

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

$$T(n) = 4T(n-2) - 2 - 1 \quad \text{--- ②}$$

Now put $n = n-2$ in ①

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

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

Put in ②

$$T(n) = 4[2T(n-3) - 1] - 2 - 1$$

$$T(n) = 8T(n-3) - 4 - 2 - 1$$

$$T(n) = 2^k T(n-k) - 2^{k-1} - 2^{k-2} = \dots - 2^0 \quad \text{--- (3)}$$

k^{th} term \rightarrow let $n-k=1$
 $k=n-1$

Put $k=n-1$ in (3)

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

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

i.e. Series in G.P.

$$a = 1/2 \quad r = 1/2$$

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

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

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

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

$T(n) = O(1)$

Q6 - int $i = 1, s = 1,$
 while ($s \leq n$) {
 $i++ ; s = s + i ;$
 $\text{print } s \text{ ("#")} ;$
 $}$

$\Rightarrow i = 1, 2, 3, 4, 5, 6, \dots$
 $s = 1 + 3 + 6 + 10 + 15 + \dots + T_{n-1} + T_n$
 $T_k = 1 + 2 + 3 + 4 + \dots + k$

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

For k iterations

$$1 + 2 + 3 + \dots + k \leq n$$

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

$$\frac{k^2+k}{2} \leq n$$

$$O(k^2) \leq n$$

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

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

Q6 - Time complexity of -

```
void function(int n){  

    int i, count=0;  

    for (i=1 ; i*i <= n ; i++)  

        count++;  

}
```

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

Q1 -

$$i^2 = n$$

$$i = \sqrt{n}$$

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

$$\sum_{i=1}^{\sqrt{n}} 1 + 2 + 3 + 4 + \dots + \sqrt{n}$$

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

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

Q7 - Time Complexity of
void function(int n)

```

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

```

For j & k

$$1, 2, 4, 8, 16, \dots, n$$

$$2^0, 2^1, 2^2, 2^3, 2^4, \dots, 2^k$$

If it is in G.P.

$$a=1 \quad r=2$$

$$t_k = a \cdot r^{k-1}$$

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

$$n = \frac{2^k}{2} + f(0 \cdot a) + f(1 \cdot a) + \dots + f(a) = O(2^k)$$

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

$$\log_2 2 + \log_2 n = \log_2 2$$

$$k = \log(n)$$

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

$$T.C. = n \times \log(n) \times \log(n)$$

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

Q8- Time complexity of

function ($\text{int } n$)

```
if ( $n == 1$ ) return;
```

```
for ( $i = 1$  to  $n$ ) {
```

```
    for ( $j = 1$  to  $n$ ) {
```

```
        printf (" *");
```

```
}
```

```
} function ( $n - 3$ );
```

```
}
```

\Rightarrow for ($i = 1$ to n)

we get $j = n$ times every time

$$\therefore i \cdot j = n^2$$

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

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

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

and $T(1) = 1$

Substitute each value in $T(n)$

$$T(n) = n^2 + (n-3)^2 + (n-6)^2 + T(n-9)$$

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

Let $n-3k=1$

$$k = \frac{n-1}{3} \quad \text{total terms} = k+1$$

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

$$T(n) \approx K n^2$$

$$T(n) \approx (K-1)/3 \cdot n^2$$

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

Q9 - Time Complexity of

void function (part n)

2

```
for(i=1 : i<n) {
    for(j=1 : j<=n ; j=j+1)
        print (*)
}
```

\Rightarrow For $i=1$ $j = 1+2+\dots+(n-1)+i$
 $i=2$ $j = 1+3+5+\dots$
 $i=3$ $j = 1+4+7+\dots$

n^{th} term of AP is

$$T(n) = a + d \times n$$

$$T(n) = 1 + d \times n$$

$$(n-1)/d = n$$

for $i=1$ $(n-1)/1$ times
 $i=2$ $(n-1)/2$ times
 $i=n-1$

we get,

$$\begin{aligned} T(n) &= i_1j_1 + i_2j_2 + \dots + i_{n-1}j_{n-1} \\ &= \frac{(n-1)}{1} + \frac{(n-2)}{2} + \frac{(n-3)}{3} + \dots + 1 \\ &= n + \frac{n}{2} + \frac{n}{3} + \dots + \frac{n}{n-1} - n \times 1 \\ &= n \left[1 + \frac{1}{2} + \frac{1}{3} + \dots + \frac{1}{n-1} \right] - n \times 1 \\ &= n \times \log n - n + 1 \\ \text{Since } \int \frac{1}{x} dx &= \log x \\ T(n) &= O(n \log n) \end{aligned}$$

Q10- For the functions, n^k and c^n , what is the asymptotic relationship between these functions?

Assume that $k \geq 1$ and $c > 1$ are constants. Find out the value of c and n_0 for which relation holds.

→ As given n^k and c^n

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

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

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

$$\forall n > n_0 \text{ & constant, } a > 0$$

$$\text{for } n_0 = 1, c = 2$$

$$1^k < a^2$$

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