Name: Ayushi Mehra

Sem: IV

Sec: G1

University Roll no.: 2016702

Class Roll no.: 32

DAA TUTORIAL: 1

1) Asymptotic notations means towards infinity. These notations tell the complexity of an algorithm when the input is very large.

Different types of Asymptotic Notations-:

1) **Big-oh (O)**

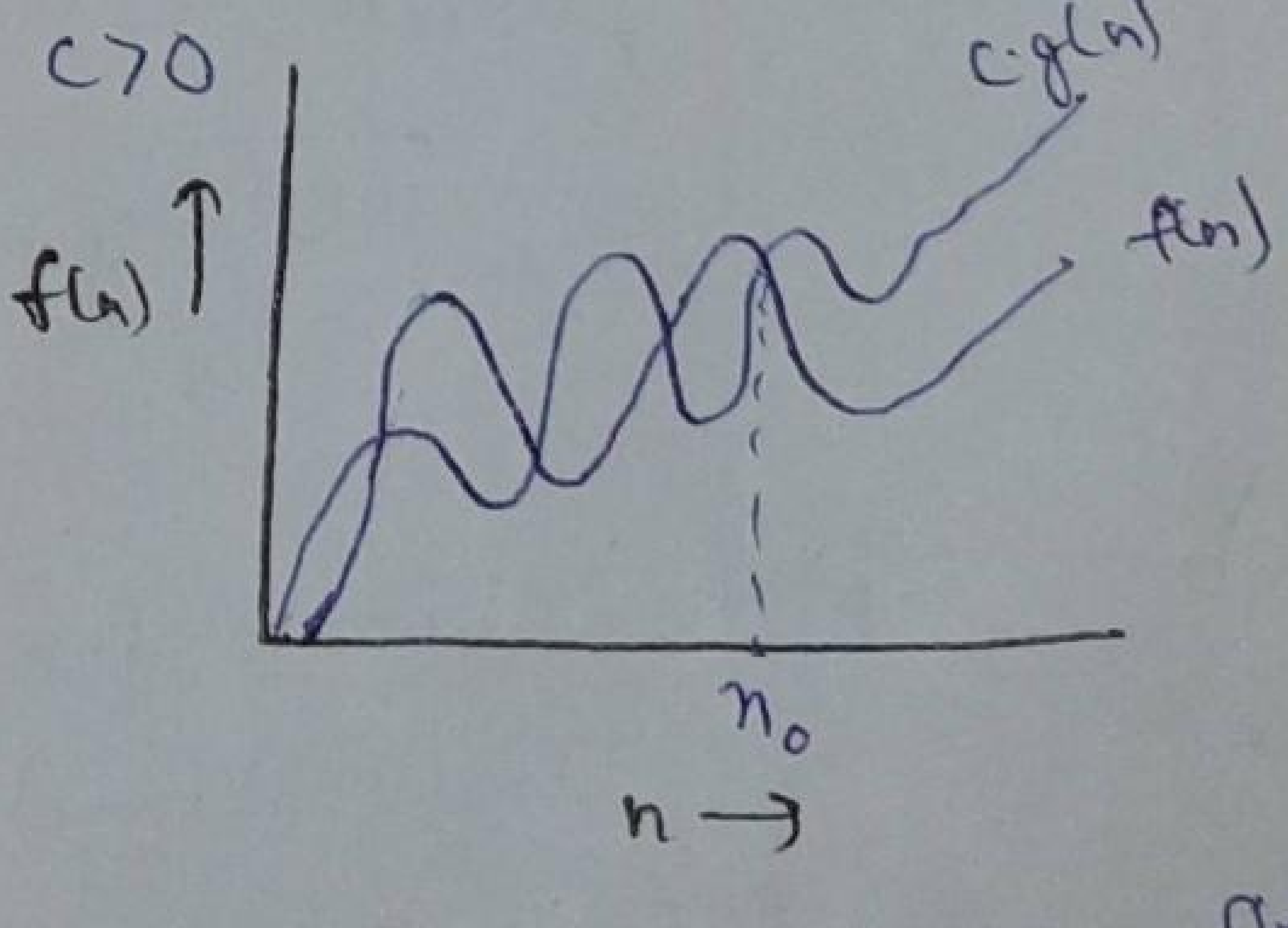
$$f(n) = O(g(n))$$

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

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

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

$\forall\ n \geq n_0$, some constant $c > 0$



2) **Big Omega (Ω)**

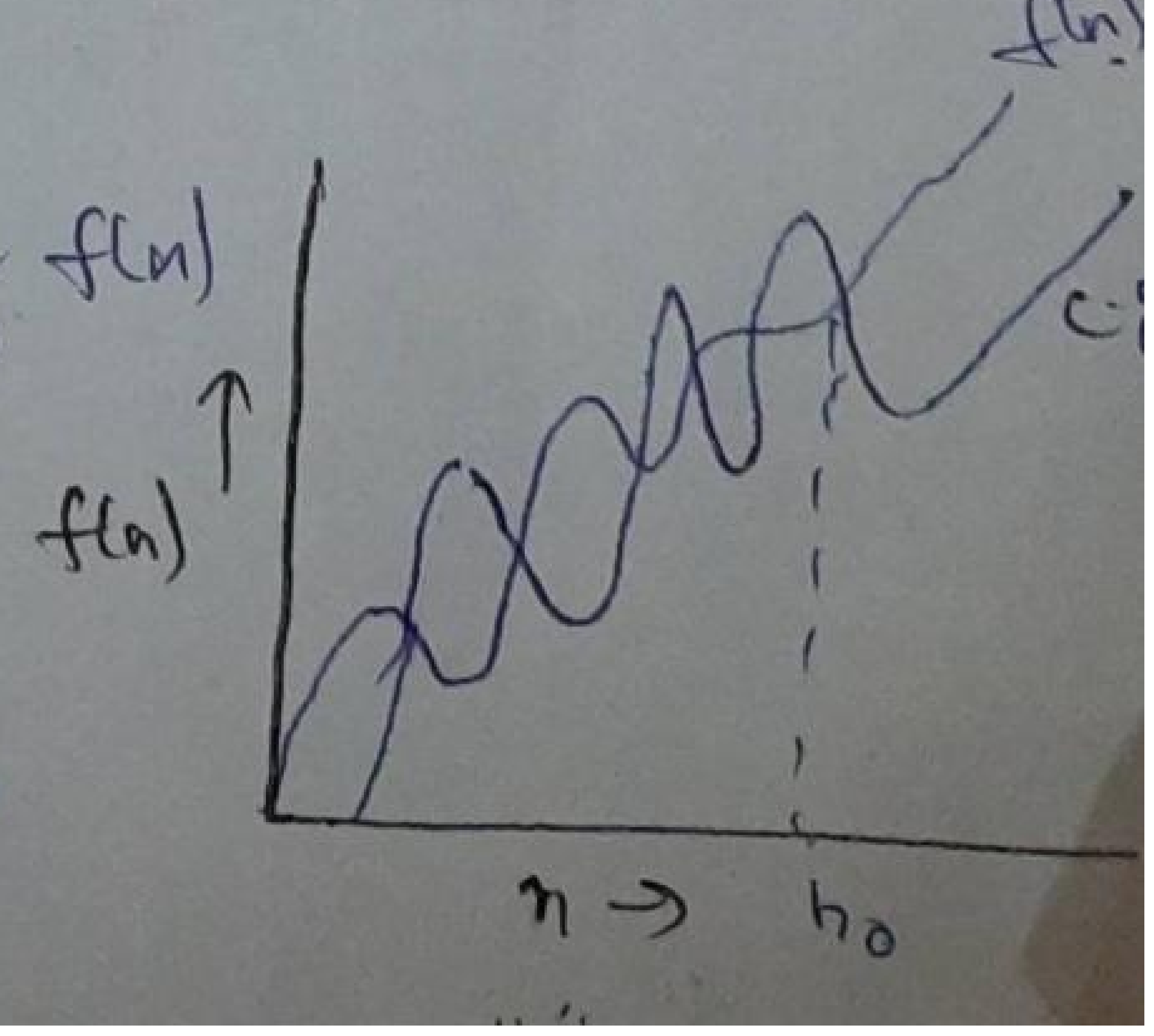
$f(n) = \Omega(g(n))$

$g(n)$ is the "tight" lower bound of $f(n)$

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

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

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



Scanned by TapScanner

(3) Theta (Θ)

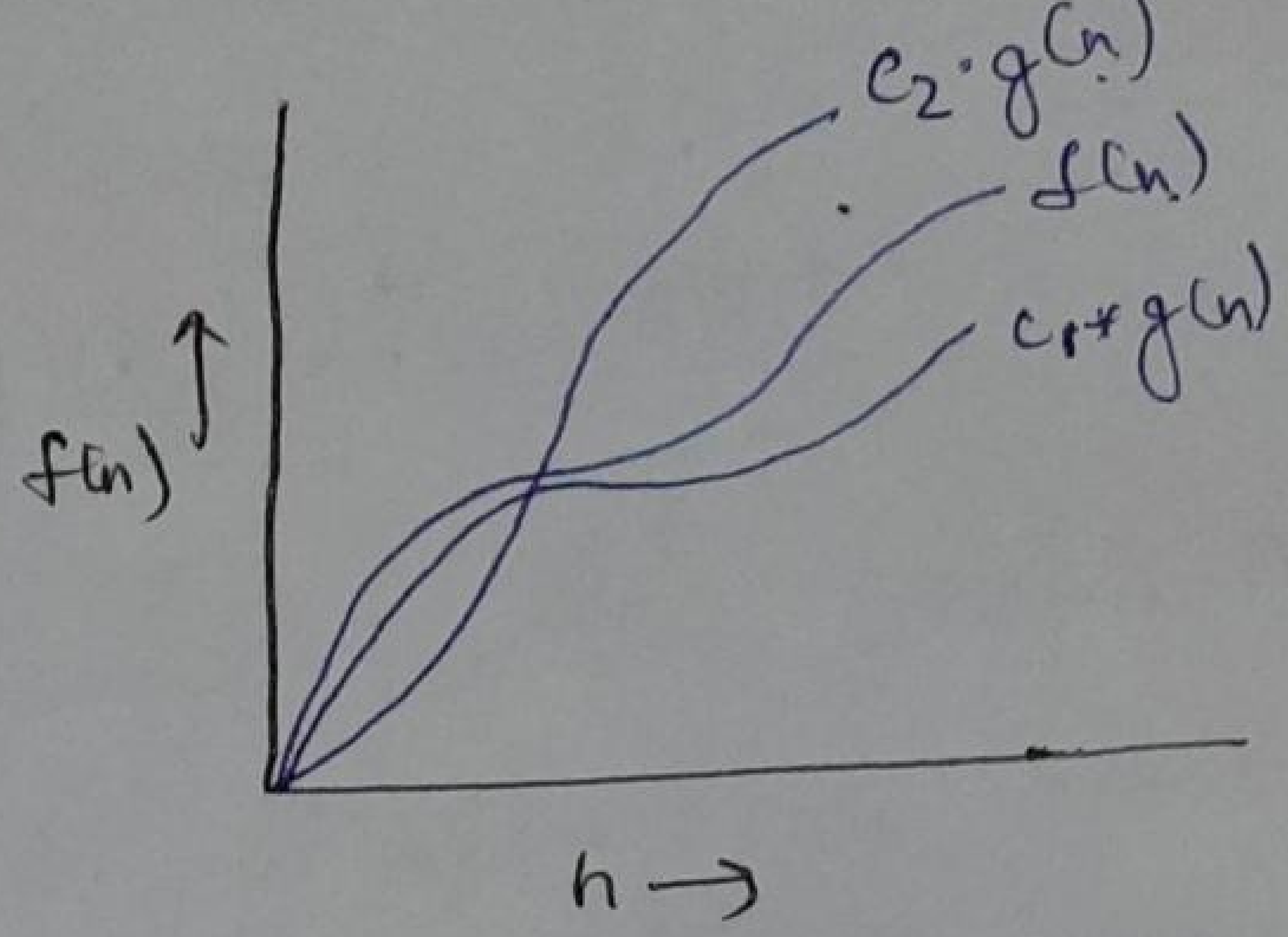
Theta gives the tight upper and lower bound both.

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

iff. $c_1 \cdot g(n) \le f(n) \le c_2 \cdot g(n)$

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

some constant, $c_1 \& c_2 > 0$.



Q2.) for (i=1 to n) { i = i*2 };

i	i
1	2
2	4
⋮	
n	2^k

$\log_2 n = \log_2 2^k$

$\log_2 n = k \log_2 2$ $\qquad (\log_a a = 1)$

$\log_2 n = k$

$\boxed{O(\log_2 n)}$ Ans

3) $T(n) = 3T(n-1)$ if $n > 0$, otherwise 1

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

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

$T(n) = 3^2\, T(n-2)$

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

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

⋮

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

$= 3^n T(0)$ $\Rightarrow$ $\boxed{O(3^n)}$ Ans

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

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

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

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

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

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

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

$= 2^3 T(n-3) - 2^2 - 2^1 - 2^0$

$= 2^n T(n-n) - 2^{n-1} - 2^{n-2} - 2^{n-3} \ldots 2^2 - 2^1 - 2^0$

$= 2^n \cdot 1 - 2^{n-1} - 2^{n-2} - 2^{n-3} \ldots 2^2 - 2^1 - 2^0$

$= 2^n - 2^n - 1$

$T(n) = 1$

$\Rightarrow \boxed{O(1)}$ Ans

5)

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

i	s
1	1
2	3
3	6
	$\vdots$
	n

23 $1+3+6+10 \ldots + \cancel{k} = \cancel{n}$

$\sum_{i=1}^{n} 1 + (1+2) + (1+2+3) + \ldots + n.$

$\sum_{i=1}^{n} \frac{n(n+1)}{2} = \frac{1}{2} \sum_{i=1}^{n} n^2 + n$

$= \frac{1}{2}\left[\frac{n(n+1)(2n+1)}{6} + \frac{n(n+1)}{2}\right]$

$$= \frac{1}{2}\left[\frac{k(k+1)}{2}\left[\frac{2k+1}{3}+1\right]\right]$$

$$= \frac{1}{2}\left[\frac{k(k+1)}{2}\left[\frac{2k+1+3}{3}\right]\right] = \frac{k(k+1)}{\cancel{4}_{2}}\frac{\cancel{2}(k+2)}{3}$$

$$= \frac{k(k+1)(k+2)}{6} = n.$$

$$= \quad k^3 = n$$

$$k = \sqrt[3]{n}$$

$$\boxed{O(\sqrt{n})} \text{ Ans}$$

6) void function (int n) {

```
            int i, count=0;
        for(i=1 ; i*i <=n ; i++)
            count++;    // O(1)
        }
```

as $i^2 <= n$

$\Rightarrow i <= \sqrt{n}$

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

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

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

$$T(n) = \frac{n+\sqrt{n}}{2} \Rightarrow \boxed{T(n) = O(n)} \text{ Ans.}$$

7) void function (int n) {

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

}
```

$for(i = \cancel{n/2}\,1 ; i<=n ; i++) = O(n/2) = O(n)$

$for(j=1; j<=n ; j=j*2) = k = \log_2 n = O(\log_2 n)$

$for(k=1; k<=n ; k=k*2) = x = \log_2 n = O(\log_2 n)$

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

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

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

8) function (int n)

```
{ if(n==1)
    return;    // O(1)
  for(i=1 to n) {  // i=1,2,3,4 ... n = O(n)
     for(j=1 to n) {  // j=1,2,3,4 ... n^2 => O(n^2)
        printf("*");
     }
  }
  function(n-3);  // T(n/3)

}
```

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

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

$c = \log_3 1 = 0$

$\Rightarrow n^c = n^0 = \cancel{1 > n^2}$

$= 1 < n^2$

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

9) void function(int n){

```
    for(i=1 to n) {
      for(j=1; j<=n ; j=j+1)
      printf("*"); // O(1)
    }
}
```

for(i=1 to n) // ~~O(n)~~ O(n)

for(j=1; j<=n; j++) // O(n)

Total complexity $\Rightarrow O(n^2)$

10) As given n^k and c^n

relation between n^k & c^n is $\rightarrow$

$n^k = O(c^n)$ as $n^k \leq ac^n$

$\forall\ n \geq n_0$ and some constant areas $a > 0$.

for, $n_0 = 1$

$c = 2$

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

$\boxed{n_0 = 1 \text{ and } c = 2}$ Ans.