DAA Tutorial -1

Mohammad Anas Khan

20

CST SPL-1

TCS 409

Tutorial -1

20021335

2017514 (Uni Roll No.)

Anas

Mohd. Anas Khan

CST SPL-1

20

Sol^n 1 Asymptotic notations are used to describe the asymptotic running time of an algorithm

It is defined in terms of functions whose domains are the set of natural numbers $N = \{0, 1, 2, \ldots\}$

These are convenient for describing the worst-case running-time function $T(n)$.

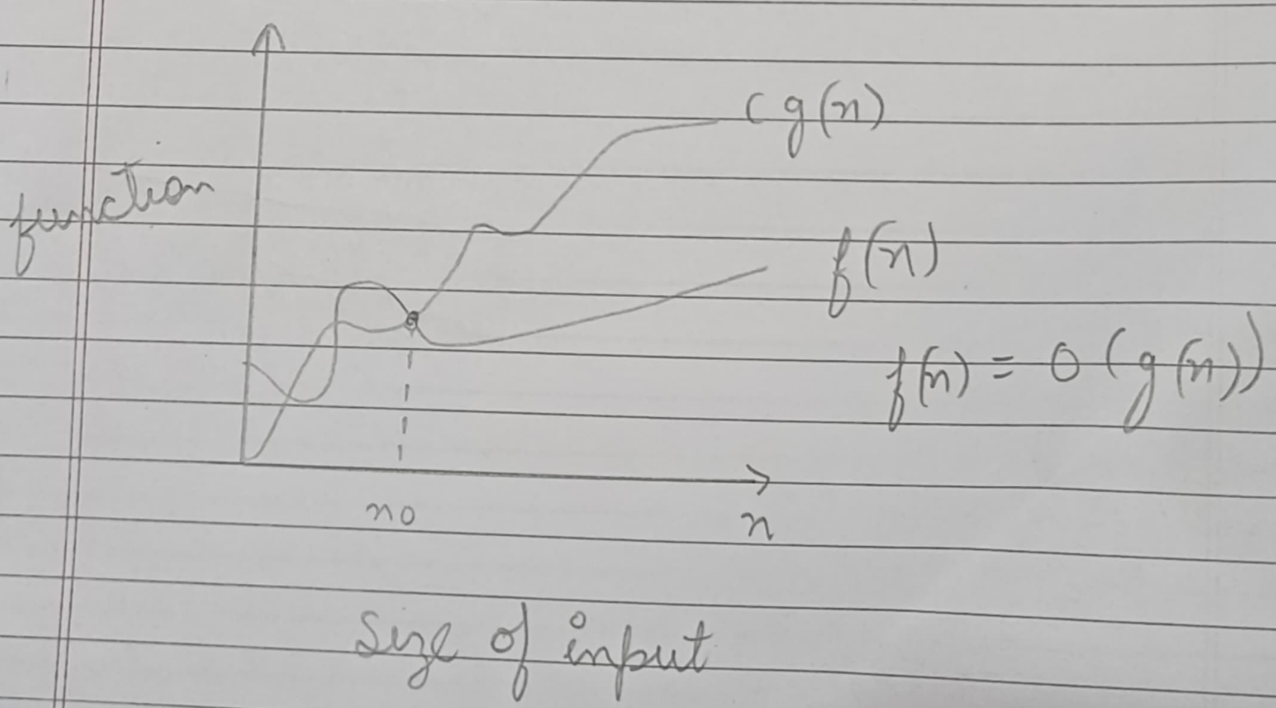
These help us finding the complexity of an algorithm when i/p is very large

Different asymptotic notations are as follows:-

1- Big O (O)

It defines an upper bound of an algorithm it bounds a function only from above

eg:- Insertion Sort $O(n^2)$



Mohd. Anas Khan
CST SPL-1
20

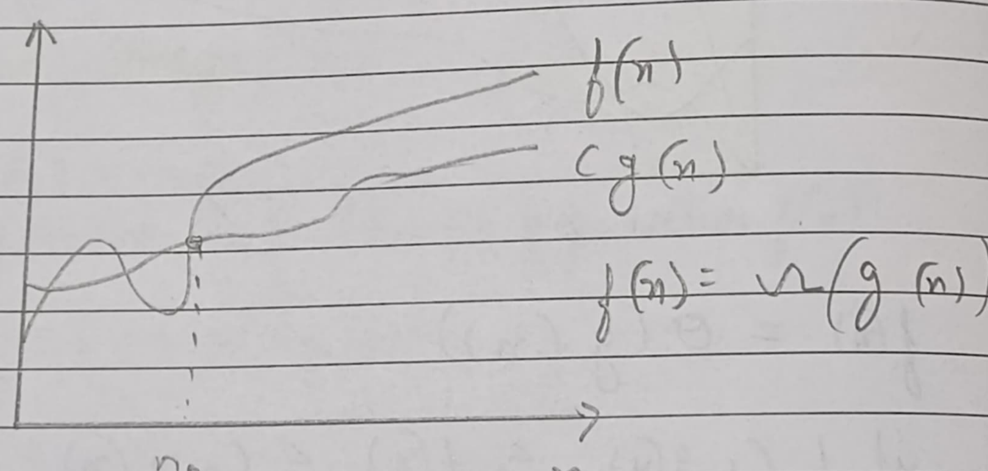
Date
Page

$f(n) = O(g(n))$ [tight upper bound]

if and only if $f(n) \le c\,g(n)$
$\forall\ n \ge n_0$

for some constant, $c > 0$

2 - Big Omega (Ω)
It provides an asymptotic lower bound. It is useful when we have a lower bound on the time complexity of an algorithm.
Eg <u>Insertion sort</u>: $\Omega(n)$



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

$g(n)$ is tight lower bound of function $f(n)$

if and only if $f(n) \ge c\,g(n)$
$\forall\ n \ge n_0$

for some constant, $c > 0$

Mohd Anas Khan

CST SPL-1

20

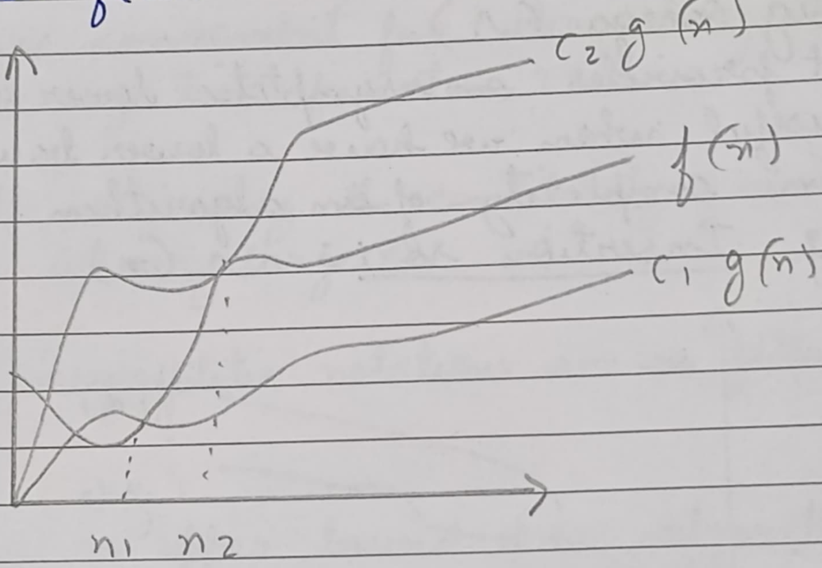
Date

Page

3- Theta (θ) It bounds a function from above & below, so it define exact asymptotic behaviour

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

g (n) is both "tight" upper & lower bound of function f(n).



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

$$\text{if } f \quad c_1 g(n) \le f(n) \le c_2 g(n)$$

$$\forall n \ge n_1, n_2$$

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

4- Small O (o)

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

g (n) is upper bound of function f(n)

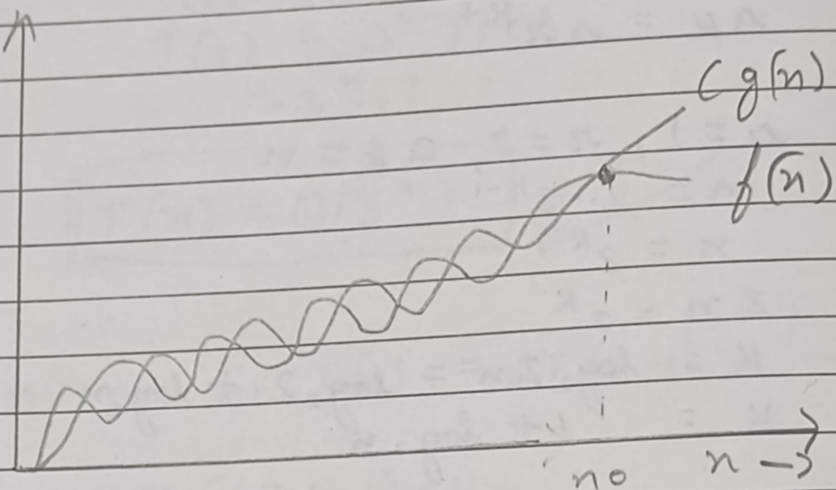
when $f(n) < c(g(n))$

$$\forall n > n_0$$

Mohd Anas Khan
CST SPL-1
20

Date
Page

and $\forall$ constants, $c > 0$



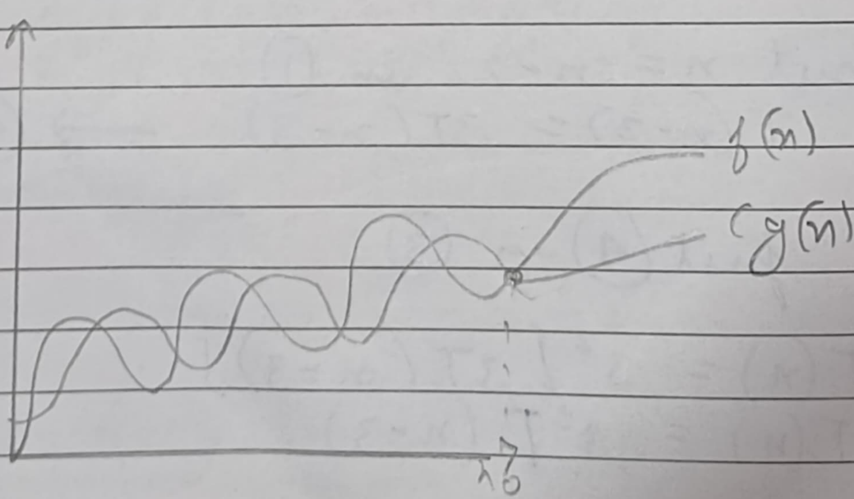
5- Small Omega (ω)

$f(n) = \omega(g(n))$
$g(n)$ is lower bound of function $f(n)$

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

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

$\forall$ $n > n_0$
and $\forall$ constants, $c > 0$



Mohd Anas Khan
CST SPL-1
20

Date
Page

sol (2)

let k terms

$i = \overbrace{1, 2, 4, 8 \ldots\ldots n}$ // G.P

$a_k = a r^{k-1}$

$a = 1,\ r = 2,\ a_k = n$

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

$n = 2^{k-1}$

$2n = 2^k$

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

$k = 1 + \log_2 n$

$T(n) = O(\log_2 n + 1) = \boxed{O(\log_2 n)}$ A

sol (3)

By backward substitution

put $n = n-1$ in $T(n) = 3T(n-1)$ — (1)

$T(n-1) = 3T(n-2)$ — (2)

put (2) in (1)

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

put $n = n-2$ in (1)

$T(n-2) = 3T(n-3) \longrightarrow$ (4)

put (4) in (3)

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

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

Mohd Anas Khan
CST SPL-1
20

Date
Page

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

Let $n-K=0$
$T(n) = 3^n T(0)$
$= 3^n \cdot 1$

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

A

Sol 4)

Put $n = n-1$ in $T(n) = 2T(n-1)-1$ — (1)

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

put (2) in (1)

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

put $n = n-2$ in $T(n) = 2T(n-1)-1 \rightarrow$ (1)
$T(n-2) = 2T(n-3)-1$ — (4)

put (4) in (3)

$T(n) = 2^2[2T(n-3)-1]-2-1$
$= 2^3 T(n-3) - 2^2 - 2^1 - 2^0 \longrightarrow$ (5)

Generalizing

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

Let $n-K=0$ $\boxed{n=K}$

Mohd Anas Khan

CST SPL-1

20

Date ______

Page ______

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

as $T(0) = 1$

$$T(n) = 2^n - 2^{n-1} - 2^{n-2} \ldots - 2^0$$
$$= 2^n - [1 + 2^1 + 2^2 \ldots 2^{n-1}]$$
$$= 2^n - \left[1 \times \frac{(2^n - 1)}{(2-1)}\right]$$

$$T(n) = 2^n - 2^n + 1$$
$$\boxed{T(n) = O(1)} \text{ Ans}$$

Sol 5

i	s	
1	1	1
2	1+2	3
3	1+2+3	6

The values in s at i^{th} iteration is the sum of first i positive integers.

so, to break out of loop.

$s > n$

i.e, if K is total iterations then sum of K terms $> n$

ie $1 + 2 + 3 + \ldots + K > n$

$$\frac{K(K+1)}{2} > n$$

$$\frac{K^2 + K}{2} > n$$

$$K^2 > n$$

Mohd Anas Khan
CST SPL-1
20

Date ____
Page ____

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

$$\boxed{T(n) = O(\sqrt{n})} \quad A$$

sol^n (6)

$i = 1, 2, 3, \ldots\ldots n$
$i^2 = 1, 4, 8, \ldots\ldots n$

$i^2 <= n$ or $i <= \sqrt{n}$

for K^{th} term : $T_K = a + (K-1)d$
$a = 1, \; d = 1$

$$T_K <= \sqrt{n}$$

$$\sqrt{n} = 1 + (K-1)1$$
$$\sqrt{n} = K$$

$$\boxed{T(n) = O(\sqrt{n})} \quad A$$

sol^n (7)

i	j	K
$n/2$	$\log n$	$\log n$
n		
n	$\log n$	$\log n$

sol

Mohd Anas Khan
CST SPL-1
20

Date
Page

$\frac{n+1}{2}$ times

$$O(j * k) = O\left(\left(\frac{n+1}{2}\right) * \log n * \log n\right)$$

$$O\left(\left(\frac{n+1}{2}\right) * (\log_2 n)^2\right)$$

$$\boxed{T(n) = O\left(n (\log_2 n)^2\right)} \quad A$$

sol 8 $T(n) = T(n-3) + n^2 \longrightarrow (1)$
$T(1) = 1 \longrightarrow (2)$

put $n = n-3$ in (1)

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

put (3) in (1)

$$T(n) = T(n-6) + (n-3)^2 + n^2 \longrightarrow (4)$$

put $n = n-6$ in (1)

$$T(n-6) = T(n-9) + (n-6)^2 \longrightarrow (5)$$

put (5) in (4)

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

Mohd Anas Khan
CST SPL-1
20

Date
Page

generalize

$$T(n) = T(n-3k) + (n-3(k-1))^2 + (n-3(k-2))^2 + \text{----} \; n^2$$

let $n - 3k = 1$

$$\frac{n-1}{3} = k$$

$$T(n) = T(1) + \left(n - 3\left(\frac{n-1}{3} - 1\right)\right)^2 + \left(n - 3\left(\frac{n-1-2}{3}\right)\right)^2 \dots$$

n^2

$$= T(1) + (n - [(n-1) - 3])^2 + (n - [(n-1) - 6])^2 + (n - (n-1-9)]^2 + \text{----} \; n^2$$

$$= 1 + [3+1]^2 + [6+1]^2 + \text{----} \; n^2$$

$$= 1 + 4^2 + 6^2 + \text{----} \; n^2$$

$n^2 + \text{----} 1$

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

sol^n (9)

i	j
1	n times
2	1+3+5+ ---+n times

$$a_n = a + (k-1)d$$

$a = 1$, $d = 2$

$$n = 1 + (k-1).2$$

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

$$k = \left(\frac{n-1}{2}\right) + 1 \Rightarrow k = \frac{n+1}{2}$$ no. of terms

Mohd Anas Khan
CST SPL-1
20

Date
Page

for $i=2$, $j = \frac{n+1}{2}$ times

for $i=3$ $\quad j = 1+4+7+ \dots n$ times

$$n = 1 + (k-1)d$$
$$n = 1 + (k-1)3$$
$$\frac{n-1}{3} + 1 = k$$
$$k = \frac{n+2}{3}; \text{ no. of terms}$$

generalising

for $i=n$ $\quad j = \frac{n+k-1}{k}$ times

$$T(n) = \underbrace{n + \frac{n+1}{2} + \frac{n+2}{3} + \dots \frac{n+k-1}{k}}_{n \text{ terms}}$$

General term $= \frac{n+k-1}{k}$

$$\sum_{i=1}^{n} \frac{n+k-1}{k} \Rightarrow \frac{\sum_{i=1}^{n} n + \sum_{i=1}^{n} k - \sum 1}{k}$$

$$\Rightarrow \frac{n\frac{(n+1)}{2} + nk - n}{k}$$

$$\Rightarrow \frac{n^2 + \frac{n}{2} + nk - n}{k}$$

Mohd Anas Khan

CST SPL-1

20

Date

Page

After removing constant terms

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

Sol^n (10) $n^k = O(c^n)$

as $n^k \leq a \cdot c^n$

$\forall \; n \geq n_0$ for some constant $a > 0$

for $n_0 = 1$

$c = 2$

$1^k \leq a 2^1$

$n_0 = 1$ & $c = 2$

Anas