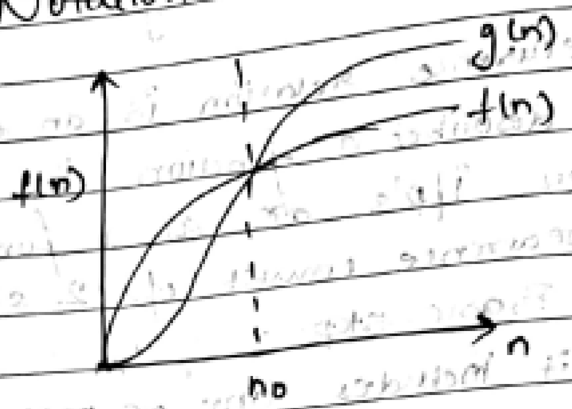
Asymtotic Notation.

Upperbond 1) Big O
Lowerbond 2) Omega &
Upper & lower 3) Theta 'Θ'



$f(n) = O\,g(n)$ if there exist some +ve constant c & no such that $f(n) \leq c \cdot g(n)$ for all $n \geq no$.

Big O. → worst → max^m no of steps → upperbond

Omega → best → min^m no of steps. → Lowerbond

Theta → avg → max^m or min. no of steps → Upper & lower

Q. Show that $f(n) = 2n^2 + n + 1$ is big $O(n^2)$.

Given $f(n) = 2n^2 + n + 1$
$g(n) = n^2$

cond^n for $O(n^2)$.

n	g(n)	f(n).
0	0	0
1	1	2+1+1 = 4
2	4	8+2+1 = 11
3	9	18+3+1 = 22
4	16	32+4+1 = 37
5	25	30+5+1 = 36
6	36	72+6+1 = 79

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

$2n^2 + n + 1 \leq c \cdot n^2$

÷ by n^2. both sides

$2 + \frac{1}{n} + \frac{1}{n^2} \leq c$

$n_0 = 1 \quad \therefore \quad \frac{2}{1} + \frac{1}{1} + \frac{1}{1} \leq c$

$$4 \leq c$$

$\therefore$ for $c = 4$ & $n_0 = 1$:

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

i.e $2n^2 + n + 1 \leq c \cdot n^2$.

let $n_0 = 1$ $c = 4$. $\quad 2 + 1 + 1 \leq 4 \cdot 1$

$$4 \leq 4$$

let $n_0 = 2$ $c = 4$ $\quad 8 + 2 + 1 \leq 4 \cdot 4$

$$11 \leq 16.$$

Q. $2n^2 + n + 1 = f(n)$. $\quad O(n^3)$

Cond^n $\quad f(n) \leq c \cdot g(n)$

$$2n^2 + n + 1 \leq n^3 \cdot c$$

$\div$ by n^3 both sides

$$\frac{2}{n} + \frac{1}{n^2} + \frac{1}{n^3} \leq c.$$

for $n_0 = 1$

$$\frac{2}{1} + \frac{1}{1} + \frac{1}{1} \leq c$$

$$4 \leq c$$

$\therefore$ for $c = 4$ & $n_0 = 1$.

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

i.e $2n^2 + n + 1 \leq c \cdot n^3$.

Scanned by CamScanner

DATE: / /

Q $f(n) = 2n^2+n+1 \quad O(n)$.

To prove: $f(n) \leq c.g(n)$

i.e

$2n^2+n+1 \leq c \cdot n$

$\div$ by n.

$2n+1+\frac{1}{n} \leq c$.

$n_0 = 1 \quad 2+1+\frac{1}{1} \leq c$

$4 \leq c$.

$n_0 = 2 \quad 4+1+\frac{1}{2} \leq c$

$\frac{11}{2} \leq 4$

$5.5 \leq 4$. (X Wrong)

$\therefore f(n) \neq O(g(n))$

$\therefore$ We can derrive a property here that funct in g(n) should have same or greater power than that of f(n) to be a O(g(n)).

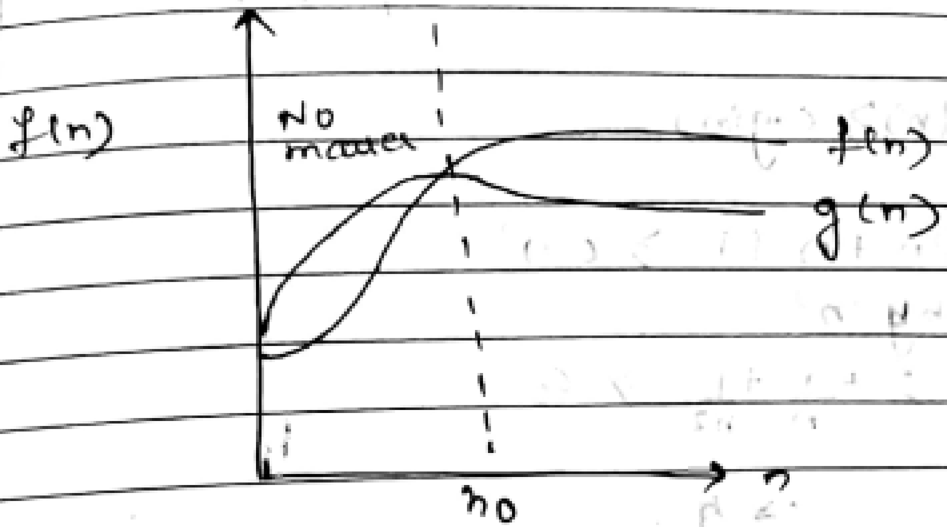
Property

$$f(n) = a_m n^m + a_{m-1} n^{m-1} \ldots + a_0 n^0$$

$a_m \neq 0$

$$f(n) = O(n^m).$$

Big Omega (Ω).



$f(n) = \Omega(g(n))$ if there exist two +ve constant c & n_0 such that

$$f(n) \geq c \cdot g(n) \quad \forall\, n \geq n_0.$$

Q. $f(n) = 2n^2 + n + 1$ show $\Omega(n)$.

$$f(n) = \Omega(n).$$
$$f(n) \geq c \cdot g(n)$$

$$2n^2 + n + 1 \geq c \cdot n$$

$\div$ by n.

$$2n + \frac{1}{n} + 1 \geq c.$$

18

for $n = 1$ $\quad c = \cancel{3}\ 4$

for $n = 1$.	~~$3 \geq 3$~~	$4 \geq 4$
$n = 2$	~~$7 \geq 6$~~	$11 \geq 8$
$n = 3$	~~$13 \geq 9$~~.	$22 \geq 12$

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

Q. $f(n) = 2n^2+n+1$ is $\Omega(n^2)$, $\Omega(n^3)$

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

$2n^2+n+1 \geq c \cdot n^2$

$\div$ by n^2

$2 + \frac{1}{n} + \frac{1}{n^2} \geq c$

for $n=1$ $c=2$.

for $n=1$ $4 \geq 2$

$n=2$ $11 \geq 8$

$n=3$ $22 \geq 18$

Hene $f(n) = \Omega(n^2)$.

for n^3. $f(n) \geq c \cdot g(n)$

$2n^2+n+1 \geq c \cdot n^3$

for $n=1$ $c=4$.

overall.

$n=1$ $4 \geq 4$

$n=2$ $11 \geq 32$ ✗

Hene $f(n) \neq \Omega(n^3)$.

Q. Consider $f(n) = 2n^3 + 3n^2 + 1$ & $g(n) = 2n^2 + 3$.
show that
1. $f(n)$ is $\Omega(g(n))$.
2. $g(n) \neq \Omega(f(n))$
3. $n^3 = \Omega(g(n))$
4. $f(n) \neq \Omega(n^4)$
5. $n^2 \neq \Omega(f(n))$.

Sol^n

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

$$2n^3 + 3n^2 + 1 \geq c \cdot (2n^2 + 3)$$

$$\frac{2n^3 + 3n^2 + 1}{2n^2 + 3} \geq c$$

$n = 1$ $c = \frac{6}{5}$ $c = 6/5$

$n = 2$ $c = \frac{29}{11}$ 2.08 $c = 6/5$ $6/5$

$n = 3$ $c = \frac{82}{21}$ $c = 6/5$

$18 + 3 = 21$

$27 \times 2 = 54$, $54 + 27 + 1 = 82$

$c = 6/5$

$\therefore c \neq \leq n.$

$n = 1$ $6 \geq 6$

$n = 2$ $29 \geq \frac{66}{5} = 13.$

$n = 3$ $82 \geq \frac{126}{5} = 25.$

Hence $f(n) = \Omega f(n)$.

(3) $g(n) \neq \Omega(f(n))$.
let us assume it to be true.

$g(n) \geq c \cdot f(n)$ — (A)

$2n^2+3 \geq c \cdot 2n^3+3n^2+1$

$\frac{2n^2+3}{2n^3+3n^2+1} \geq c$.

for $n=1$ $c = 5/6$.

$\therefore$ for inequality (A)

$n=1$ $\cancel{5/6 \geq 5/6}$ $5 \geq 5$

$n=2$ $\cancel{11/29 \geq}$ $11 \geq \frac{5 \times 29}{6} = \frac{145}{6} = 24$

$n=3$ $21 \geq \frac{5 \cdot 82}{6} = \frac{5 \cdot 41}{3} = \frac{205}{3} = 68$

Hence $g(n) \neq \Omega f(n)$.

Other way

$2n^2+3 \geq c \cdot 2n^3+3n^2+1$
ignore const
$2n^2 \geq c \cdot 2n^3+3n^2$
ignore n^2 in front of n^3

$\cancel{2}n^2 \geq c \cdot \cancel{2}n^3$

$\therefore \frac{1}{n} \geq c$

as $n \uparrow$ $c \downarrow$ $\therefore$ inequality decreses.

(iii) $n^3 = \Omega(g(n))$

i.e

$n^3 \geq c.(2n^2+3)$ — (A)

$\cancel{n^3 \geq 2n^2 c}$ $\frac{n^3}{2n^2+3} \geq c.$ $\frac{1}{2}.5$

$\cancel{\frac{n}{2} \geq c}$.

$n=1$ $c = 1/5$

18

in Inequality (A) $n=1$ $1 \geq 1/.$

$n=2$ $8 \geq 11/5 = 2.$

$n=3$ $27 \geq 21/5 = 4.$

Hence $n^3 = \Omega(g(n))$.

(iv) $f(n) \neq \Omega(n^4)$

Let us assume.

$f(n) \geq c.n^4.$

$2n^3+3n^2+1 \geq c.n^4$

$\frac{2}{n} + \frac{3}{n^2} + \frac{1}{n^4} \geq c.$

for $n=1$ $c = 6$.

for inequality. $n=1$ $c = 6 \geq 6$

$n=2$ $29 \geq 32$

$n=3$ $82 \geq$ ___

Hence $f(n) \neq \Omega(n^4)$.

(v) $n^2 \neq \Omega(f(n))$

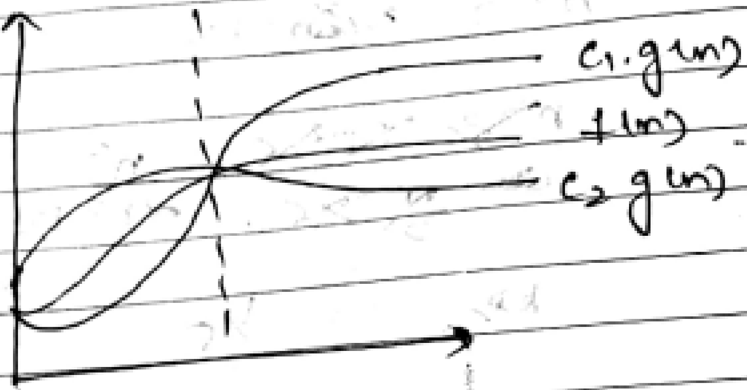
assume

$n^2 \geq c_1\ 2n^3+3n^2+1$

$n^2/2n^3+3n^2+1$ $1/n \geq c$

Scanned by CamScanner

Theta (Θ).



$f(n) = \Theta(g(n))$ if there exist c_1, c_2

$c_1 \cdot g(n) \leq f(n) \leq c_2\, g(n) \quad \forall\ n \geq n_0.$

Q. $f(n) = 2n^2 + n + 1$ $\quad g(n) = 2n^2 + 3$
show that $f(n) = \Theta(g(n))$

i.e $\quad c_1 \cdot g(n) \leq f(n) \leq c_2 \cdot g(n).$

$c_1 \cdot (2n^2+3) \leq 2n^2+n+1 \leq c_2 (2n^2+3)$

break
$c_1 (2n^2+3) \leq 2n^2+n+1 \quad — ①$

$2n^2+n+1 \leq c_2 (2n^2+3) \quad — ②$

So $\quad c_1 \leq \dfrac{2n^2+n+1}{2n^2+3} \leq c_2.$

$n = 1 \quad c_1 = \dfrac{4}{5} = 1.2.$

$5c_1 \leq 4$ $\quad\quad$ $4 \leq c_2 \cdot 5$

$c_1 = \frac{4}{5}$

At $n = 2$

$\frac{4}{5}(2 \cdot 2^2 + 3) \leq 2 \cdot 2^2 + 2 + 1$

$\frac{4}{5}(8+3) \leq 8+2+1$

$\frac{44}{5} \leq 11$

Hence $f(n) \neq \Theta(g(n))$.

Q. $f(n) = 2n^3 + 3n^2 + 1$ & $g(n) = 2n^2 + 3$.

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

(2) $f(n) \neq \Theta(n^2)$ $\quad\quad$ $n^2 \leq$

(3) $n^4 \neq \Theta(g(n))$ $\quad\quad$ $\frac{1}{n} \leq$

Sol^n

$c_1(2n^2+3) \leq 2n^3 + 3n^2 + 1 \leq c_2(2n^2+3)$ $\quad\quad$ 6

acc
potentia

Amortized Analysis

Q What is amortized complexity? Explain the amortized complexity for 4 bit binary no. from 0-8. Write algo for binary incrementation operation.

overall flips		
0	0000 ←	see at most every step one bit sets to 1 & more than one go to zero.
1	000$\boxed{1}$	
3	00$\boxed{1}$0	
4	001$\boxed{1}$	
7	0100 ←	
8	0101	
10	0110	
11	0111 ←	
15	1000	

→ complexity overall.

cost

ALGO

Bin-counter (A)

1. i=0
2. while i<A.length & A[i] = 1
3. A[i] = 0
4. i++
5. i<A.length
6. A[i]=1

Here complexity for n operations is same since average is taken.

Here n operation sequence is considered.

Eg stack operation. push & multipop.

3 types of Amortized Analysis

(1) Aggregate (2) Accounting (3) Potenti

n = no. of operation.
T(n) = Total cost.

avg cost (amortized cost) = $\frac{T(n)}{n}$.

push $O(1) \times n/2 = n/2 = O(n)$

pop $O(1) \times n/2 = n/2 = O(n)$

$O(n) + O(n) = \frac{2O(n)}{n} = \frac{2\cancel{n}}{\cancel{n}} = 2$

$= O($

→ multipop (S, k)

1. while k>0 && top != -1
2. Pop(S)
3. k = k-1;

multipop is denoting multiple pop ope

Aggregation will always return a co complexity O(1)

x	3 2 1 0 (Array)
0	0000
1	000$\boxed{1}$
2	00$\boxed{10}$
3	001$\boxed{1}$
4	0$\boxed{100}$
5	010$\boxed{1}$
6	01$\boxed{10}$
7	011$\boxed{1}$
8	$\boxed{1000}$
	$\frac{n}{8}\ \frac{n}{4}\ \frac{n}{2}\ n$

Scan R to L. contn.

→ change all 1's to 0

&

→ 1st zero to 1.

& stop

Note: The funcn down will be performed on 0 array.

ALGORITHM

BIN_COUNTER (A)

1. i=0
2. while A[i]=1 AND i<A.length
   A[i]=0
   i++
3. if i<A.length
   A[i]=1.

A[0] = n

A[1] = n/2

A[2] = n/4.

A[i] = i → 0, 1, 2 .. (k-1)

A[i] = $\frac{n}{2^i}$

Total cost = $n+\frac{n}{2}+\frac{n}{2^2}$ ....

$=\sum_{i=0}^{k-1}\frac{n}{2^i}$

$GP = n\sum_{i=0}^{\infty}\left(\frac{1}{2}\right)^i$

$= 2n$

Total complexity (worst) $= O(n)$.

Avg. complexity = O(1).

potential to be zero.

→ difference in actual cost is there for every operation (+, -) any.

(2) Accounting Method [~~loop me credit karna~~]

$c' \geq c$

c' → amortized cost, c → total cost

c' → should be equal or greater than actual cost.

stack	Actual cost	Amortized (cost)
push	O(1)	2
pop	O(1)	0
multipop	min(s, k)	0

diff b/w amortized & Actual is credit

Eg. cafeteria. Push +2. Pop -1. push total = credit ~~potential~~

if amortized > Actual.

$\sum_{i=1}^{n} c_i' \geq \sum_{i=1}^{n} c_i$ [overall c' > c (should be) not on a particular step.]

operation	cost	pay	use	reserve
0 → 1	+2	2	-	2
1 → 0	-1	2	1	3
		2	-	5
		2		7 = 0

see problem here was while we credit karna charge & rest the decrease hona chahiye

Accounting for a particular object — like on plate

∴ 0 → 1 -1, 1 → 0 +2.

0 → 3. -1, +2, -1, -1 = -1. H(4). -1 +2 +2 -1 = 2.

[Def: Pay change in Energy.] **Potential Method** [Like physics → energy derived from other energies]

$$\sum_{i=1}^{n} \hat{C_i} = \sum_{i=1}^{n} \left(C_i + \phi(D_i) - \phi(D_{i-1}) \right)$$

$1-2+2-3 \ldots -n+n$

$$= \sum_{i=1}^{n} \left(C_i + D_n - D_0 \right).$$

$$\hat{C_i} = C_i + \phi(D_i) - \phi(D_{i-1})$$

5, 6 **Push** $\hat{C_i} = C_i + \phi(6) - \phi(5)$

$= 1 + 1 = 2$

Pop $\hat{C_i} = C_i + \phi(5) - \phi(6)$

$= 5 - 6 = -1$

Push → store's energy for further operation

Pop → uses stored Energy.

Amortized $O(1)$ cost, total $= O(n)$

Actual $= O(n)$. we know.

[But should be upper bound.]

[Potential for overall. Database] Like stack