AKSHIT
1815009

01 December 2019
Sunday
Wk 48 • 335-030

M	T	W	T	F	S	S
30	31					1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

DEC '19

ASSIGNMENT

Q1

Big O	Big omega	Theta Notation
1) It is like <= rate of growth of an algorithm	1) It is like >= rate of growth is greater or equal	1) It is like == meaning the rate of growth is equal to specified value
2) The upper bound of Algorithm is represented by Big O	2) The Algorithm lower bound is represented by Big omega	2) The bonding of function from above and below is represented by θ
3. Big O (O) worst case.	3. Big Ω best case	3) Big Theta θ is Avg case

Q2

Ans Analysis of Algorithm is the process of analyzing the problem solving capacity of algorithm in terms of the memory and the time required for execution of Algorithm

Worst case - The maximum number of steps required for any test case.

Important Calls ✓ Things to Do ✓ Meetings ✓

M	T	W	T	F	S	S
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

JAN '20

1815004

December 2019

Wk 49 • 336-029

02 Monday

Best Case :- The minimum time or space required for a test case

Average Case :- The Avg time and space required for any test case.

Q3

Ans

$$\lim_{n\to\infty} \frac{f(n)}{g(n)} = \lim_{n\to\infty} \frac{\log(n^2)}{\log(n+3)}$$

$\left(\frac{\infty}{\infty}\right)$ form

Using L Hopital Rule

$$\lim_{n\to\infty} \frac{\frac{1}{n^2}(2n)}{\frac{1}{n}} = 2 \text{ const}$$

But greater than 0 and Pre

$\therefore$ $f(n)$ is $\Theta(g(n))$

Important Calls ✓

Things to Do ✓

Meetings ✓

03 December 2019
Wk 49 • 337-028
Tuesday

18/50

M	T	W	T	F	S	S
30	31					1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

DEC '19

(ii)

9.00

Ans $f(n) = n$ $g(n) = \log n^2$

10.00

$g(n) = 2\log n$

11.00

Ignoring constants for comparing complexities

12.00

1.00

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

2.00

$f(n) > \log(n)$

3.00

$\therefore$ growth rate of $f(n)$ is high as compare to $g(n)$

4.00

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

5.00

(iii)

6.00

$f(n) = n$, $g(n) = \log^2 n$

7.00

if we put $n = 2^8$

$f(n) = 2^8$ $g(n) = \log^2(2^8)$

$f(n) = 256$

$= \log 2^8 * \log 2^8$

$= 64$

Important Calls ✓

Things to Do ✓

Meetings ✓

☐

☐

☐

Hence growth of $f(n)$ is greater than $g(n)$

9.00

(iv)

10.00 $f(n) = n \log n + n$; $g(n) = \log n$

11.00 By Putting $n = 2^8$

12.00 $f(n) = 2^8 (\log 2^8) + 2^8$ $\quad$ $g(n) = \log 2^8$

$= 256 \times 8 + 256$ $\quad$ $= 8$

1.00 $= 2304$

2.00 Hence growth of $f(n)$ is more than $g(n)$

3.00 $f(n) = \Omega(g(n))$

4.00

5.00 (v) $f(n) = 10$, $g(n) = \log 10$

6.00 As both $f(n)$ and $g(n)$ are constants

7.00 $f(n) = O(g(n))$

(vi) $f(n) = 2^n$, $g(n) = 10n^2$

$f(n) = 2^n$ $\quad$ $g(n) = 10n^2$

Important Calls ✓ Things to Do for $n = 5$ ✓ Meetings

$f(n) = 32$ $\quad$ $g(n) = 250$

05 December 2019
Thursday
Wk 49 • 339-026

As growth rate of $f(n)$ is more then $g(n)$
Hence
$$f(n) = \Omega(g(n))$$

(vii)
$f(n) = 2^n$ $g(n) = 3^n$

Put $n = 5$
$f(n) = 32$ $g(n) = 243$

As growth of $g(n)$ is greater then $f(n)$
Hence $f(n) = O(g(n))$

Q
Ans

Priori Analysis	Posteriori Analysis
1 Priori Analysis is an absolute analysis.	1 Posteriori Analysis is a relative analysis.
2 It is independent of languages, types of compiler	2. It is dependent on types of language and compiler.
3 It will give appropriate answer.	3 It will give exact answer.
4 It uses asymptotic notation.	4 It does not uses asymptotic notation.

Important Calls ✓ Things to Do ✓ Meetings ✓

Q5

9.00 Ans

1 $(3/2)^n$ exponential

10.00

2

11.00 1 constant

12.00 3 $(3/2)n$ linear

1.00 4 $2n^3$ polynomial

5 2^n exponential

2.00

Q6

3.00 Ans let $f(n) = n^3 \log_2 n$

4.00 $g(n) = 3n \log_e n$

5.00

$g(n) = 3n \log_8 n$

6.00 $= \frac{3n}{3} \log_2 n = n \log_2 n$

7.00

let $n = 8$

$f(8) = (8)^3 * \log_2 8$ $g(8) = 8(\log_2 8)$

$= (8)^3 * 3$ $= 8 \times 3$

Important Calls Things to Do Meetings

DEC '19

M	T	W	T	F	S	S
30	31					1
2	3	4	5	6	7	8
9	10	11	12	13	14	15
16	17	18	19	20	21	22
23	24	25	26	27	28	29

Hence $f(n)$ is much more growing then $g(n)$

9.00

10.00

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

11.00

Q7

Ans

$$f(n) = 10n^2 + 7$$

12.00

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

1.00

For proving Big O for $f(n)$ we have to find c and n_0

2.00

$$10n^2 + 7 \leq c(n^2) \;\forall\; n \geq n_0$$

3.00

By inspection c must be greater than 10 so taking $c = 11$

4.00

And if we take $n_0 = 3$

5.00

then $f(3) \leq 11 g(3)$

08 Sunday

$$10n^2 + 7 \leq c(n^2)$$

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

Q2

Ans

let $g(n) = n$

Important Calls

Things to Do

Meetings

If we assume $c = 7$ and say $n \geq 1$ then we can say

$f(1) \geq c\,g(1)$

9.00

$f(1) \geq 27g(1)$

10.00

Then we can say.

11.00

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

12.00

$27n^2 + 16n + 27 \geq 27(n)$

1.00

Q8

2.00 Ans (1) $f(n) = \Theta(f(n))$

3.00 $f(n) = O(f(n))$

4.00 $f(n) = \Omega(f(n))$

5.00 (2) Symmetric

$f(n) = \Theta(g(n))$. if and only if

6.00 $g(n) = \Theta(f(n))$

7.00 (3) Transitivity

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

imply $f(n) = \Theta(h(n))$

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

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

Important Calls ✓ Things to Do ✓ Meetings ✓

DEC '19	M	T	W	T	F	S	S
	30	31					1
	2	3	4	5	6	7	8
	9	10	11	12	13	14	15
	16	17	18	19	20	21	22
	23	24	25	26	27	28	29

9.00 $m(n) = m(n-1) + 1 + m(n-1)$

10.00 $m(n) = 2m(n-1)+1$. for $n > 1$

11.00 $m(1) = 1$

12.00 $m(n) \begin{cases} m(1) & : n = 1 \\ 2m(n-1)+1 & : n > 1 \end{cases}$

1.00 $m(n) = 2m(n-1) + 1$ —(i)

2.00 $m(n-1) = 2m(n-2) + 1$ —(ii)

3.00 $m(n-2) = 2m(n-3) + 1$ —(iii)

4.00 Put (ii) in (i)

5.00 $m(n) = 2(2m(n-2)+1)+1$

6.00 $m(n) = 2^2 m(n-3) + 2^2 + 2 + 1$

7.00 $m(n) = 2^k m(n-k) + 2^{k-1} + 2^{k-2} + 2^{k-3} + \ldots 2 + 1$

Important Calls	✓	Things to Do	✓	Meetings	✓
	☐		☐		☐
	☐		☐		☐
	☐		☐		☐

Base condition

$$n - k = 1$$
$$k = n - 1$$

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

$$m(n) = \frac{2^n}{2} + 2^n - 1$$

growth determining function is 2^n

$$m(n) = O(2^n)$$

Q11
Ans

$$F(n) = F(n-1) + 1 \quad \text{—(i)}$$
$$F(n-1) = F(n-2) + 1 \quad \text{(ii)}$$
$$F(n-2) = F(n-3) + 1 \quad \text{(iii)}$$
$$F(n) = F(n-3) + 1 + 1 + 1$$
$$F(n) = F(n-k) + k$$

$$n - k = 0$$
$$k = n$$

Important Calls

Things to Do

Meetings

$$F(n) = F(0) + n$$
$$F(n) = 1 + n$$
$$\therefore F(n) = O(n)$$

Q 12

for $i \leftarrow 0$ to $n-1$ do

for $j \leftarrow 0$ to $n-1$ do

$C[i, j] \leftarrow 0.0$

for $k \leftarrow 0$ to $n-1$ do

$C[i, j] \leftarrow C[i, j] + A[i, k] * B[k, j]$

~~worst~~

Time complexity

$$C_{worst}(n) = \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} \sum_{k=0}^{n-1} 1$$

$$= \sum_{i=0}^{n-1} \sum_{j=0}^{n-1} (n-1-0+1)$$

$$= \sum_{i=0}^{n-1} n(n-1-0+1)$$

$$= n^2 (n)$$

$$= n^3$$

$$C_{worst}(n) = O(n^3)$$