CS & IT ENGINEERING

Algorithms

Analysis of Algorithms

Lecture No.- 07

By- Aditya sir

Recap of Previous Lecture

Topic Small Notations

PUB PLB

(o, ω)

Topic Properties of Asymptotic Notation

Topic Problem Solving

Topics to be Covered

Topic Problem Solving with ASN ☆ (PYQ, Practice)

Topic Framework for Analysing Non-Recursive algorithm

Topic : Asymptotic Notations & Apriorist Analysis

State True / False (HW)

1. $100n.\log n = O(n.\log n)$ → T
2. $2^{n+1} = O(2^n)$: → T
3. $2^{2n} = O(2^n)$ → F
4. $0 < x < y$ then $n^x = O(n^y)$: → T
5. $(n+k)^m \neq \theta(n^m)$ $(k, m) > 0$ → F
6. $\sqrt{\log n} = 0(\log \log n)$ → F
7. $\log(n)$ is $\Omega(1/n)$ → T
8. 2^{n2} is $0(n!)$ → F
9. n^2 is $O(2^{2\log n})$ → T

10. $a^n \neq O(n^x)$, $a > 1$, $x > 0$ → T

expo, polynomial

11. $2^{\log_2 n^2}$ is $O(n^2)$ → T

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

$\Rightarrow O(n^2 \log_2 2)$

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

① $100 * n * \log n = O(n \log n) \longrightarrow$ True

② $2^{(n+1)} = O(2^n) \longrightarrow$ True

$\downarrow$

$2^{(n+1)} = 2^n * 2^1 = 2 * 2^n = O(2^n)$

$\boxed{2^{2n} \neq 2 * 2^n}$

③ $2^{(2n)} = O(2^n) \longrightarrow$ False

$\boxed{(4)^n > (2^n)}$

$\hookrightarrow 2^{(2n)} = (2^2)^n = (4)^n$

(4) $0 < x < y \implies n^x = O(n^y)$ (polynomial)

eg $\rightarrow$ 2 4

$n^2 < n^4$

$n^2 = O(n^4)$

(5) $(n+k)^m \neq O(n^m)$, $k, m > 0$ False

$(n+2)^2 \Rightarrow n^2$

Polynomial of degree m.

⑥ $\sqrt{\log_2 n} = O(\log(\log n)) \longrightarrow$ False

$\longrightarrow (\log n)^{1/2}$ $\qquad\qquad$ $\log(\log n)$

c

$\frac{1}{2} \times \log(\log n) > \log(\log(\log n))$

⑦ $\log n = \Omega\left(\frac{1}{n}\right) \longrightarrow$ T

$\log n > \frac{1}{n}$

8) $2^{(n^2)} = O(n!) \approx n^n$

False

Take log on both sides

$\log\left(2^{(n^2)}\right)$ vs $\log(n^n)$

$n^2 * \underbrace{\log_2 2}$ vs $n * \log n$

n^2 vs $n\log n$

n^2 vs $n\log n$

$\cancel{n} * n$ $\quad$ $\cancel{n} * \log n$

$n > \log n$

11 $2^{\log_2 n^2} = O(n^2) \longrightarrow$ True

$n^{2 \log_2 2}$

$= n^2 \longrightarrow O(n^2)$

Topic : Asymptotic Comparisons

1. $f(n), g(n)$: are functions
& $f(n) = O(g(n))$ — given

a) Is $f(n) = O(f(n)^2)$ → False

b) $2^{f(n)} = O(2^{g(n)})$ → False

eg2 $f(n) = (1/n)$

$(f(n))^2 = (1/n)^2 = \frac{1}{n^2}$

$\frac{1}{n} > \frac{1}{n^2}$

$f(n) \neq O((f(n))^2)$

1. Is $f(n) = O((f(n))^2)$?

eg1 :- $f(n) = n$

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

$n < n^2$

$f(n) < (f(n))^2$

$f(n) = O((f(n))^2)$ ✓

(2) given, $f(n), g(n) \longrightarrow$ functions

(2) $\boxed{f(n) = O(g(n))}$

False

Check :- $2^{f(n)} = O\left(2^{g(n)}\right)$

eg1 :- $f(n) = n$

$g(n) = n^2$

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

$\rightarrow 2^f = 2^n$

$2^g = 2^{n^2}$

$2^n \leq 2^{n^2}$

$2^n = O(2^{n^2}) \rightarrow$ True

eg2 :- $f = 2n$

$g = n$

given $f = O(g)$

$2^f = 2^{2n} = 4^n$

$2^g = 2^n$

$4^n = O(2^n)$? $\longrightarrow$ False

[MCQ]

#Q. Which one of the following statements is True for all $\boxed{\text{positive}}$ functions f(n)? PYQ

↳ gate 2022

A. $f(n^2) = \theta(f(n)^2)$, when f(n) is a polynomial

B. $f(n^2) = O(f(n)^2)$ → small o

C. $f(n^2) = O(f(n)^2)$, when f(n) is an $\boxed{\text{exponential}}$ function

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

(A) $f(n^2) = \Theta\left((f(n))^2\right)$, $f(n) \longrightarrow$ polynomial

eg: $f(n) = n^3$

$\longrightarrow f(n^2) = (n^2)^3 = n^{2\times 3} = n^6$

$\longrightarrow (f(n))^2 = (n^3)^2 = n^{3\times 2} = n^6$

$$\boxed{(n^a)^b = (n^b)^a}$$

True

(B) $f(n^2) = O\left((f(n))^2\right)$

eg:- $f(n) = n$

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

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

small o

$n^2 = o(n^2) \longrightarrow \boxed{\text{False}}$

False ← (c) is $f(n^2) = O((f(n))^2)$, if $f(n)$ is expo ?

$2^{(n^2)}$ $2^{(2n)}$ eg: $f(n) = 2^{(n)}$

$f(n^2) = 2^{(n^2)}$

$2^{(n^2)} \neq 2^{(2n)}$

$\log(2^{n^2})$ $\log(2^{2n})$ $(f(n))^2 = (2^n)^2 = 2^{2n}$

$2^{n^2} > 2^{2n}$

$n^2 > 2n$

(D) is $f(n^2) = \Omega((f(n))^2)$? $\longrightarrow$ False

eg2

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

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

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

$\log(n^2)$ vs $(\log n)^2$

$2 * \log(n) < (\log n) * (\log n)$

$(\log n) + (\log n) \quad (\log n) * (\log n)$

eg1

$f(n) = n^2$

$f(n^2) = (n^2)^2 = n^4$

$(f(n))^2 = (n^2)^2 = n^4$ ✓

Topic : Adding Functions

The sum of two functions is governed by the dominant one, namely:

$O(f(n)) + O(g(n)) \rightarrow O(\max(f(n), g(n)))$

$\Omega f(n)) + \Omega(g(n)) \rightarrow \Omega(\max(f(n), g(n)))$

$\theta(f(n)) + \theta(g(n)) \rightarrow \theta(\max(f(n), g(n)))$

eg:-

$f = n^2 \longrightarrow O(f) = O(n^2)$

$g = n^3 \longrightarrow O(g) = O(n^3)$

$f+g \rightarrow n^2+n^3$

$O(f+g) = O(n^2+n^3) \rightarrow O(\max(n^2, n^3))$

$O(n^3)$

n^3

Topic : Multiplication of Functions

$O(f(n)) * O(g(n)) \rightarrow O(f(n) * g(n))$

$\Omega(f(n)) * \Omega(g(n)) \rightarrow \Omega(f(n) * g(n))$

$\theta(f(n)) * \theta(g(n)) \rightarrow \theta(f(n) * g(n))$

$\theta(f(n)) * \theta(g(n)) = \theta\left(f(n) * g(n)\right)$

eg:- $f = n^2 \longrightarrow O(n^2)$

$g = n^3 \longrightarrow O(n^3)$

$f \times g = n^2 \times n^3$

$= n^5$

$\downarrow$

$O(n^5) \longrightarrow O(f \times g)$

Topic : ~~Adding Functions~~ Asymptotic Comparisons

1. $f(n) = n, g(n) = \log n$ → $\log n < n$

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

2. $f(n) = n^2 \log n, g(n) = n\log^{10} n$

3. $f(n) = n^3, 0 < n \leq 10{,}000$

   $= n, n > 10{,}000$

   $g(n) = n, 0 < n \leq 100$

   $= n^3, n > 100$

Q.3 Two package are availabe for processing a Data base having 10^x records. Package A takes a time of $\boxed{10.n.\log n}$ while package B takes a time of $0.0001n^2$ for processing 'n' records. Determine the $\boxed{\text{smallest integer x}}$ for which package 'A' $\boxed{\text{outperforms}}$ Pacakge 'B'.

$\boxed{x = 6}$ ✓

outperforms → performs better → (takes lesser time)

② $f = n^2(\log n)$ vs $g = (n \log^{10} n)$

$\log^{10} n = (\log n)^{10}$

$$n^2(\log n) \quad \text{vs} \quad n * (\log n)^{10}$$

$$\cancel{(n * \log n)} * n \quad \text{vs} \quad \cancel{(n * \log n)} * (\log n)^9$$

$$\boxed{\text{Poly} > \log}$$

$$n \quad \text{vs} \quad (\log n)^9$$

$$\log(n) \quad \text{vs} \quad \log((\log n)^9)$$

$$\log(n) \quad \text{vs} \quad 9 * \log(\log n)$$

Ans:- $f(n) = \Omega(g(n))$

③ $f(n) = n^3, \quad 0 \le n \le 10{,}000$

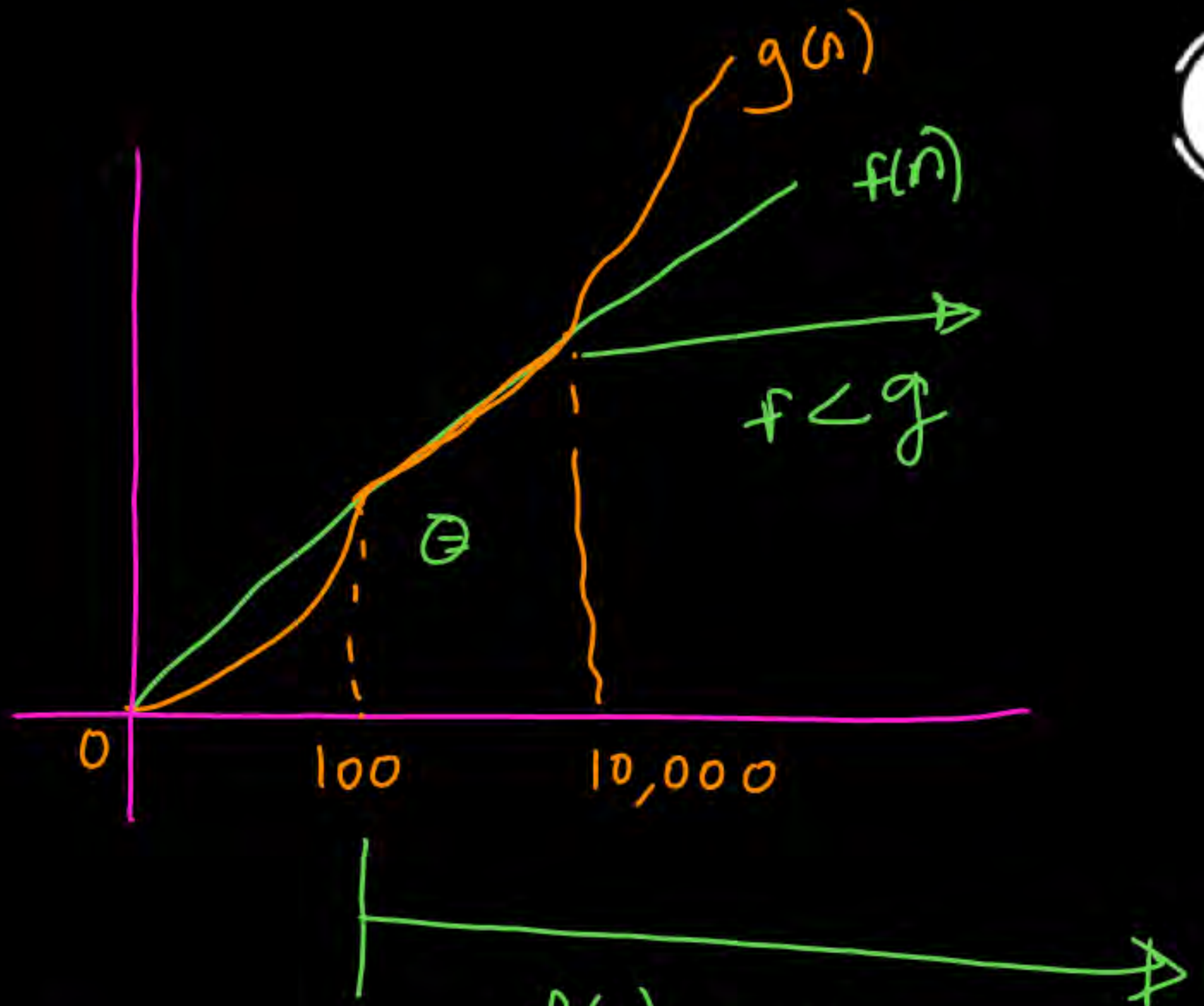
$n, \quad n > 10{,}000$

$g(n) = \begin{cases} n, & 0 < n \le 100 \\ n^3, & n > 100 \end{cases}$

$g(n) = \begin{cases} n, & 0 < n \le 100 \\ n^3, & 100 < n \le 10{,}000 \\ n^3, & n > 10{,}000 \end{cases}$

$<$

$\le$ ✓

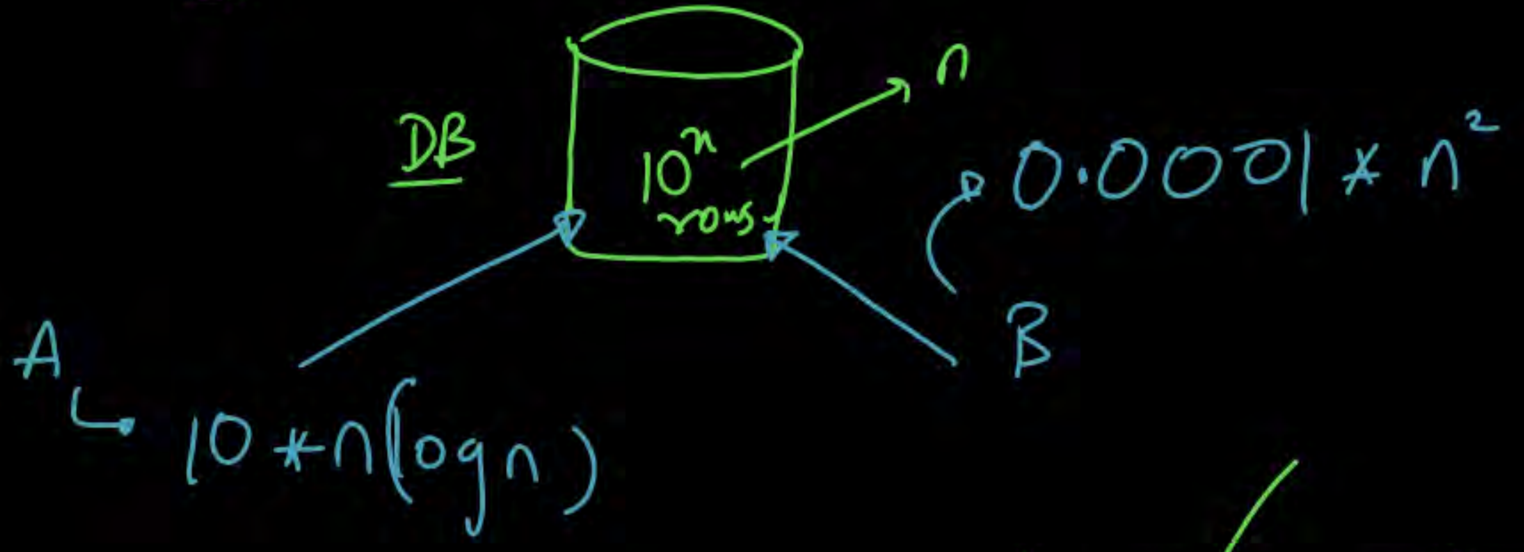


$f(n) \le g(n)$

① $f(n) = O(g(n)), \quad n > 100 \; (n_0)$

② $f(n) = O(g(n)), \quad n > 10{,}000$

4



Case :- if $x = 2 \Rightarrow \boxed{n = 10^2}$

(10^2) $A \rightarrow 10 * 10^2 * \log_{10} 10^2$

$10^3 * 2 \times 1$

$= 2 * 10^3$ units

$B \rightarrow 10^{-4} * n^2$

$= 10^{-4} * (10^2)^2$

$= 10^{-4} * 10^4$

$= 1$ unit

if $x = 3 \longrightarrow n = 10^3$

$A \rightarrow 10 \times n \log n$

$= 10 \times 10^3 \log_{10} 10^3$

$= 10^4 \times 3$

$B \longrightarrow 10^{-4} \times n^2$

$= 10^{-4} \times (10^3)^2$

$= 10^{-4+6}$

$= 10^2$ units

$\underline{A > B}$

Right approach :-

$n = 10^{x}$

$A \rightarrow 10 * n \log n$, $B \rightarrow 10^{-4} * n^2$

$$10 * 10^x * \log 10^x < 10^{-4} * (10^x)^2$$

$$x * 10^{x+1} * \cancel{\log_{10} 10}^{\,1} < 10^{-4} * 10^{2x}$$

$$x \times \cancel{10^x} * 10 < 10^{-4} * \cancel{10^x} * 10^x$$

$$x * 10 < 10^{-4} * 10^x$$

$$x < \frac{10^{-4} \times 10^x}{10}$$

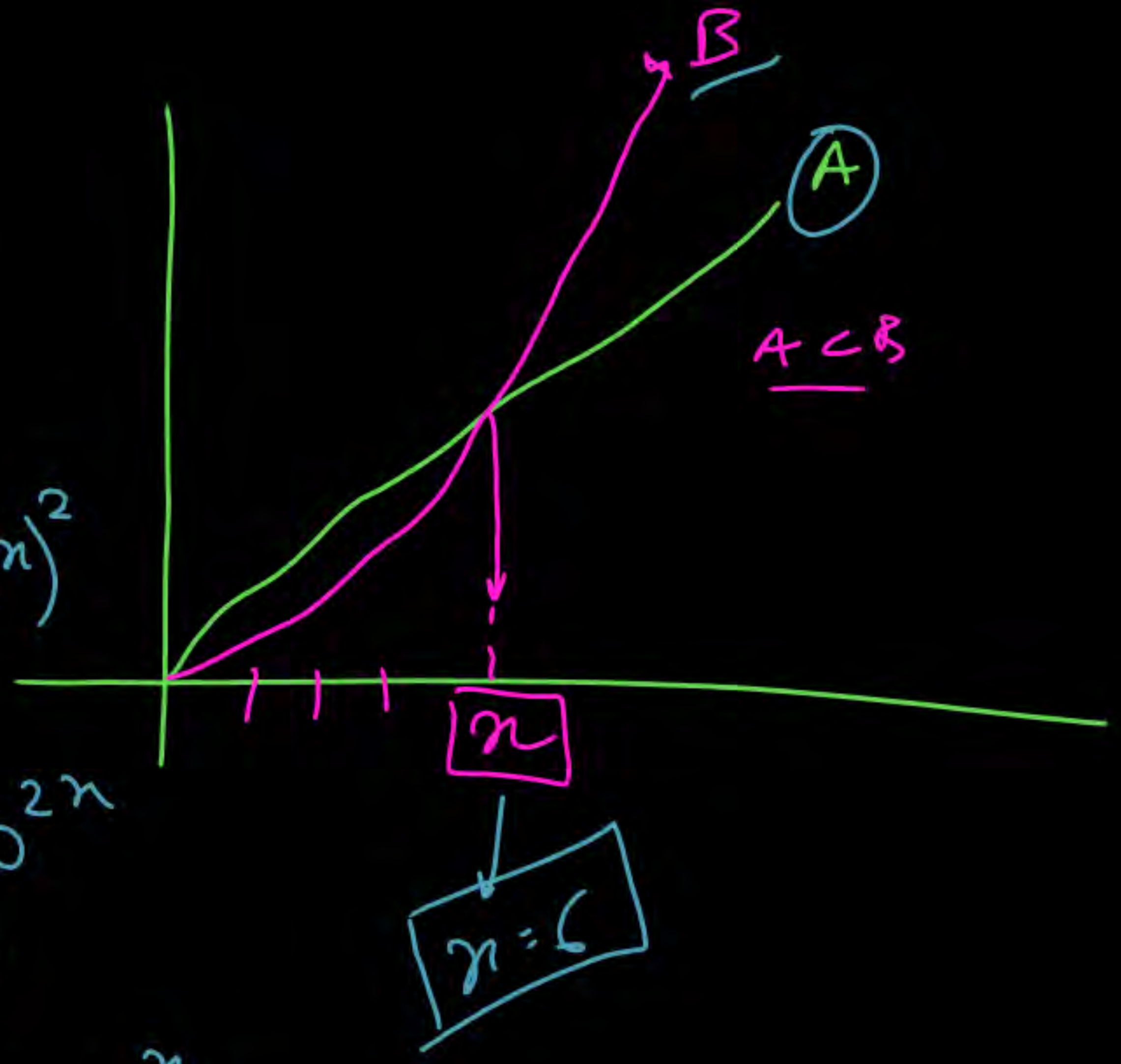
$$x < \frac{10^x}{10^5}$$

$$x < 10^{(x-5)}$$

if $\boxed{x = 6}$ → ans

$$6 < 10^{6-5}$$

$$\boxed{6 < 10}\ \checkmark$$



Topic : Asymptotic Comparisons

$\cancel{n}\sqrt{n}$ vs $\cancel{n}\log n$

$\sqrt{n} > \log n$

$\cancel{n} \times \log n$ vs $\cancel{n} \times \sqrt{n}$

0.4 Arrange the functions in increasing order of rates of growth.

1. n^2; n logn; $n\sqrt{n}$; e^n; n; 2^n; $(1/n)$
2. 2^n; $n^{3/2}$; nlogn; $n^{\log n}$
3. $n^{(1/3)}$; e^n; $n^{7/4}$; $n\log^9 n$; 1.001^n

① $\boxed{\frac{1}{n} < n < n\log n < n\sqrt{n} < n^2 < 2^n < e^n}$

① $n^2 \to P$, $n\log n \to P$, $n\sqrt{n} \to P$ ($n^{1+1/2} = n^{3/2}$), $e^n \to E$, $n \to P$, $2^n \to E$, $1/n \to P$

② $2^n \to E$, $n^{3/2} \to P$, $n\log n \to P$, $n^{\log n} \to E$

$\boxed{n\log n < n^{3/2} < n^{\log n} < 2^n}$

Ans

2^n vs $n^{\log n}$

$n \log_2 2$ vs $\log n * \log n$

③ $n^{1/3}$, e^n, $n^{7/4}$, $n\log^9 n$, $(1.001)^n$

$n^{1/3} \to P$ ✓

$e^n \to E$

$n^{7/4} \to P$ ✓

$n\log^9 n \to n(\log n)^9 \to P$ ✓

$(1.001)^n \to E$

$n^{1/3} = n^{0.3} < n^1$

$n^{7/4}$ vs $n\log n$

$n^{1+3/4}$

$= \cancel{n} * n^{3/4}$ vs $\cancel{n}\log n$

$n^{3/4}$ vs $\log n$

$n^{3/4} > \log n$

$$\boxed{n^{1/3} < n\log(n) < n^{7/4} < (1.001)^n < e^n}$$

[MCQ]

#Q. Consider the following functions from positive integers to real number:

$10, \sqrt{n}, n, \log_2 n, \frac{100}{n}$

c ✓ ↑p ↑p ↑p → p (Dec)

The correct arrangement of the above functions in $\boxed{\text{increasing order}}$ of asymptotic complexity is:

$$\frac{100}{n} < 10 < \log_2 n < \sqrt{n} < n$$

A $\log_2 n, \frac{100}{n}, 10, \sqrt{n}, n$

B $10, \frac{100}{n}, \sqrt{n}, \log_2 n, n$

C ✓ $\frac{100}{n}, 10, \log_2 n, \sqrt{n}, n$

D $\frac{100}{n}, \log_2 n, 10, \sqrt{n}, n$

[MCQ]

#Q. Which of the following is TRUE.

H.W

1. $f(n)$ is $O(g(n))$
2. $g(n)$ is NOT $O(f(n))$
3. $g(n)$ is $O(h(n))$
4. $h(n)$ is $O(g(n))$

A. $f(n)$ is $O(h(n))$

B. $h(n) \neq O(f(n))$

C. $f(n) + h(n)$ is $O(g(n) + h(n))$

D. $f(n).g(n) \neq O(g(n).\ h(n))$

[MCQ]

#Q. $f(n) = 2^n; g(n) = n^n$

HW

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

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

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

D None of these

[MCQ]

#Q. $f(n) = n.2^n; g(n) = 4^n$

HW

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

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

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

D None of these

[MCQ]

#Q. Let w(n) and A(n) represent respectively, the worst case and average case running time of an algorithm with input size of n, Which is always TRUE?

HW

A. $A(n) = o(w(n))$

B. $A(n) = \theta(w(n))$

C. $A(n) = \Omega(w(n))$

D. $A(n) = O(w(n))$

E. $A(n) = \omega(w(n))$

Topic : Take-Home Lesson:

The Big Oh notation and worst-case analysis are tools that greatly simplify out ability to compare the efficiency of algorithms.

$3n^2 - 100n + 6 = O(n^2)$ because I choose $c = 3$ and $3n^2 > 3n^2 - 100n + 6$

$3n^2 - 100n + 6 = O(n^3)$ because I choose $c = 1$ and $n^3 > 3n^2 - 100n + 6$ when $n > 3$;

$3n^2 - 100n + 6 \neq O(n)$, because for any c I choose $cn < 3n^2$ when $n > c$;

Topic : Take-Home Lesson:

$3n^2 - 100n + 6 = \Omega(n^2)$ because I choose $c = 2$ and $2n^2 < 3n^2 - 100n + 6$ when $n > 100$

$3n^2 = 100n + 6 \neq \Omega(n^3)$, because I choose $c = 3$ and $3n^2 - 100n + 6 < n^3$ when $n > 3$

$3n^2 - 100n + 6 = \Omega(n)$ because for any c I choose $cn < 3n^2 - 100n + 6$ when $n > 100c$

$3n^2 - 100n + 6 = \theta(n^2)$ because both O and apply;

$3n^2 - 100n + 6$ ne $\theta(n^3)$, because only O applies;

$3n^2 - 100n + 6$ ne $\theta(n)$, because only & Ω applies.

[NAT]

#Q. An element in an Array is called Leader if it is greater than all elements to the right of it. The time complexity of the most efficient algorithm to print all Leaders of the given Array of size 'n' is _____.

Start in next lec → (Coding based)

2 mins Summary

Topic Problem Solving with ASN * (Practice , PYQ)

Topic Framework for Analysing Recursive algorithm → Next lec

THANK - YOU