CS & IT ENGINEERING

2024

Algorithms

Analysis of Algorithms

DPP - 02

Discussion Notes

By- Aditya sir

[MSQ] → multiple can be correct

#Q. Which of the following notation is/are transitive but not reflexive

Ans:- (c, d)

A. Big oh (O) ✗

B. Big omega (Ω) ✗

C. Small oh (o) ✓

D. Small omega (ω) ✓

	O	Ω	Θ	o	ω
Reflexive	✓	✓	✓	✗	✗
Transitive	✓	✓	✓	✓	✓

[MSQ]

#Q. If $f(n) = \sum_{i=1}^{n} i^3$

Then which of the following choices is/are true for f(n)?

A. $\theta(n^4)$ ✓

B. $\Omega(n^4)$ ✓

C. ~~$\theta(n^5)$~~ $O(n^5)$ ✓

D. $\Omega(n^3)$ ✓

Tight LB

$\Omega(n^4)$

$\theta(n^4) \rightarrow O(n^4) \rightarrow$ Tight UB

$O\begin{pmatrix} n^5 \\ n^6 \\ n^7 \end{pmatrix}$ Loose UB

$\Omega\begin{Bmatrix} n^3 \\ n^2 \\ n \\ \sqrt{n} \end{Bmatrix}$ Loose LB

Ans:- (A, B, C, D)

$$f(n) = \sum_{i=1}^{n} i^3$$

$$= 1^3 + 2^3 + 3^3 \; \cdots \; n^3$$

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

$$= \frac{n^2(n+1)^2}{4} = \frac{n^2(n^2+2n+1)}{4}$$

$$f(n) = \frac{n^4+2n^3+n^2}{4} \checkmark$$

$O(n^4)$ ✓

$\Omega(n^4)$ ✓

$\Theta(n^4)$ ✓

[MCQ]

#Q. Consider the following program:

```
main ( )
{
    P = n!
    for (i = 1; i ≤ n ; ++i)
  for (j = 1 ; j ≤ P ; 2*j )
        C = C + 1;
}
```

What is the time complexity of above code?

$P \Rightarrow n^n$

$O(n)$

$n! \approx$

$n \times (n-1) \times \cdots 1$

$\approx n^n$

A. $O(n^2)$

B. $O(n^2 \log n)$

C. $O(n \log n)$

D. $O(n)$

Ans: B

for(j=1; j≤ p ; j = 2*j)

$1 \rightarrow 2^1 \rightarrow 2^2 \rightarrow 2^3 \; . \; . \; 2^k$

$\downarrow$

p

$2^k = p$

$k = (\log_2 p)$

Total TC $\rightarrow$ $O(n * \log_2 p)$

$\log_2 p = \log_2 (n^n)$

$= \underline{\underline{n \log n}}$

TC $\rightarrow O(n \times n \log n) = O(n^2 \log n)$

[MCQ]

#Q. Consider the following code:

```
main ( )
{
    i = 1; j = 1
    while (j ≤ n )
    {
        ++ i;
        j  = j + i;
    }
```

What is the time complexity of above code?

Ans :- B ✓

A. $\theta(n)$ ✗

B. $\theta\left(\sqrt{n}\right)$ ✓

C. $\theta(\log)$ ✗

D. $\theta(n\log(\log n))$ ✗

initially $\rightarrow$ 1st iter $\rightarrow$ 2nd itr 3rd itr

$i=1$ $i=2$ $i=3$ $i=4$

$j=1$ $j=1+2$ $j=1+2+3$ $j=1+2+3+4$

kth iter

$i=k+1$

$j \rightarrow 1+2+3+4 \dots (k+1)$

$\Downarrow n$

$$\sum_{i=1}^{k+1} i = n$$

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

$$k^2 \approx O(n)$$

$$\boxed{k = O(\sqrt{n})}$$

[MCQ]

#Q. Consider the following code:

```
Algorithm T(n)
{
        if (n = 1) return;
        else
{
        T(n/2);
        }
    }
```

What is the space complexity of above code?

Ans : A

A. $\theta(\log n)$

B. $\theta(n)$

C. $\theta(n\log(\log n))$

D. $\theta(\sqrt{n})$

Space Complexity → Auxiliary Space
(Additional) Space.

Recursive → Recursion Stack.

↓

Space Complexity

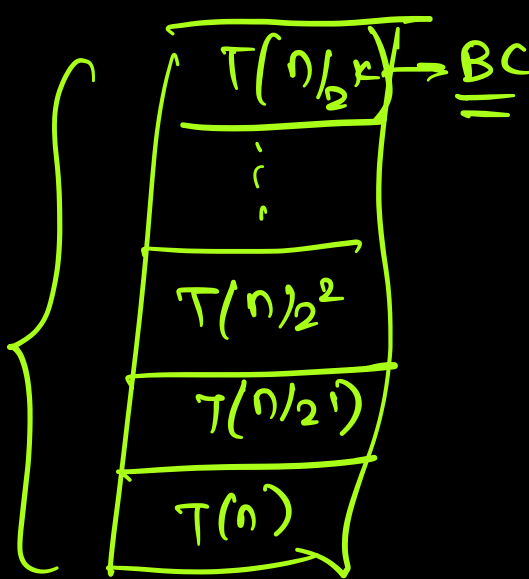
$T(=)\quad T(n) = T(n/2) + a$

$= T(n/2^2) + 2a$

$\vdots$

$$\boxed{T(n) = \log_2(n)}$$

Recursin stack



Size of Recursion Stack = 'K'

For Base Condition (To stop Recursion)

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

$2^k = n$

$$\boxed{k = \log_2 n}$$

$O(K)$

[MSQ]

Asymptotic Comparison.

#Q. $f(n) = 2^{n^2}, \quad g(n) = n! \quad h(n) = 2^{\log n^2}$

Which of the following is/are correct?

Ans: A, C

A. f(n) = Ω (g(n)) ✓

B. h(n) = Ω (g(n)) ✗

C. h(n) = O (g(n)) ✓

D. g(n) = Ω (f(n)) ✗

Conclusion $\boxed{f > g > h}$

(A) $f = \Omega(g)$
$f \geq g$ ✓

(B) $h = \Omega(g)$
$h \geq g$ ✗

(C) $h = O(g)$
$h \leq g$ ✓

(D) $g = \Omega(f)$
$g \geq f$ ✗

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

$g(n) = n! \simeq n^n$

$h(n) = 2^{\log_2 n^2}$

$= n^2$

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

$\log_2(2^{n^2}) \quad \log_2(n^n)$

$n^{\cancel{2}} \quad \cancel{n} \log n$

$n > \log n$

$g = n^n \quad h = 2^{\log_2 n^2} = n^2$

$n^n > n^2 \quad g > h$

Conclusion $\boxed{f > g > h}$

[NAT]

#Q. Consider the following notations:

1. $\sqrt{\log n} = O(\log\log n)$ → False.

2. $\log n = \Omega\left(\frac{1}{n}\right)$ → true

3. $n^2 = \theta\left(2^{2\log n}\right)$ → True

4. $(0.061)^n = \theta(1.02)^n$ → False

How many notations is/are correct? ______________.

Asymptotic Comparison.

$(0.061)^n \overset{<}{=} \theta(1.02)^n$

$\frac{1}{2} \rightarrow \left(\frac{1}{2}\right)^2 \rightarrow \frac{1}{4}$

① $\sqrt{\log n} = O(\log(\log n))$ X

<u>False</u>

$(\log n)^{1/2} \leq C * \log(\log n)$

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

Take log both sides

$\frac{1}{2}\log(\log n) \qquad \log(\log(\log n))$

$\log(\log n) = x$

$\frac{x}{2} > \log(n)$

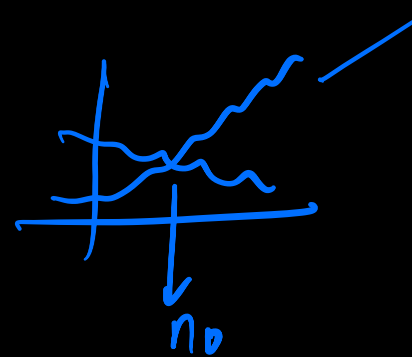
② $\log n = \Omega(1/n)$

$\underline{\log n} \geq \frac{C*1}{n}$ $\quad$ $1/n \rightarrow \underline{\underline{\text{Decr}}}$

$\downarrow$

<u>incr.</u>

$\hookrightarrow$ <u>True</u>



③

(3) $n^2 = \Theta(2^{2\log n})$

$n^2 \underset{A}{\approx} 2^{2\log n}$

Take log on both sides

$\log_2(n^2)$ $\qquad$ $\log_2(2^{2\log n})$

$2 * \log_2 n$ $\qquad$ $2\log_2 n * \cancel{\log_2 2}^{\,1}$

$2\log_2 n = 2\log_2 n$ ✓

log

$2^n = 4^n$

$(2^2)^n$

2^{2n}

$(2^n)^2$

$\cancel{2^n} \qquad \cancel{2^n} \times 2^n$

$1 \qquad 2^n$

[MCQ]

#Q. Consider the following functions:

$f_1 = 2^n$

$f_2 = n!$

$f_3 = n^n$

$f_4 = e^n$

What is the correct increasing order of above function?

A $f_1\ f_4\ f_2\ f_3$ ✓

B $f_2\ f_1\ f_4\ f_3$ ✗

C $f_2\ f_4\ f_1\ f_3$ ✗

D $f_2\ f_1\ f_4\ f_3$ ✗

Ans :- A

$f_1 \to 2^n$

$f_2 \to n!$

$f_3 \to n^n$

$f_4 \to e^n$

Imp $n! < n^n$

$n(n-1)(n-2)\cdots 1 < n \times n \times n \cdots$

$n! < n^n$

n^n

$e \simeq 2.7$

$n^n > n! > (e^n, 2^n)$

$(2.7)^n > 2^n$

$n^n > n! > e^n > 2^n$

$f_3 > f_2 > f_4 > f_1$

$f_1 \quad f_4 \quad f_2 \quad f_3$

THANK - YOU

→ Practice
→ lectures
→ Backlog clear!