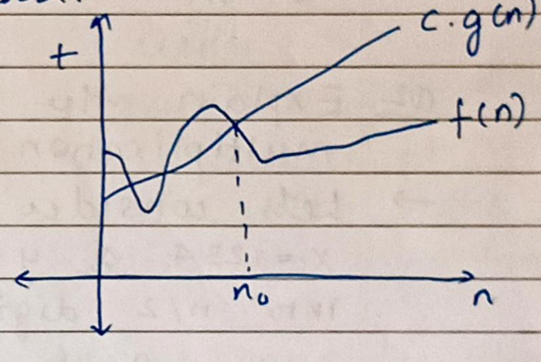
NAME: PRERNA SUNIL JADHAV
SAP ID: 60004220127
AOA ASSIGNMENT 01

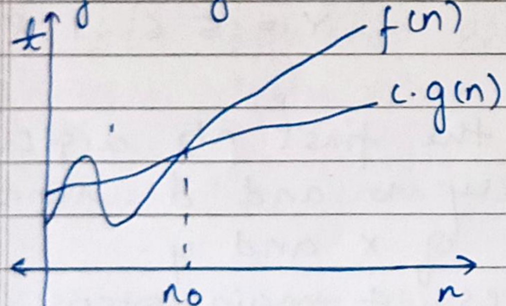
Q1 Explain asymptotic notations. Define order of growth. List various efficiency classes with example.

→ Asymptotic notation are used to describe the running time of an algorithm.

- Big Oh
  → They are used to describe the least upper bound running time
  → worst time complexity
  → $f(n) \le c\,g(n)$

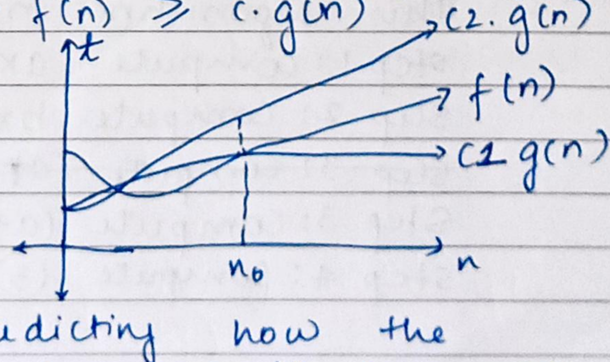


- Big omega



→ Big omega notation are used to describe the greatest lower bound running time
→ Best time complexity
→ $f(n) \ge c.g(n)$

- Theta (Θ)
  → They are used to describe average time
  → $c1.g(n) \le f(n) \le c2.g(n)$



→ Order of growth is predicting how the execution time of a program as i/p increases.

→ Efficiency classes

1. constant
2. log n - Logarithmic
3. n ~~log n~~ - Linear
4. n log n - Linearithmic
5. n^2 - Quadratic
6. n^3 - cubic
7. 2^n - Exponential
8. n! - factorial

Q2 Explain the Karatsuba algorithm for multiplication of Long Integer.

→ Lets consider two integer X and Y where X = 1234 & y = 5678. we divide the n-digit no. into n/2 digit no.s

$$X = \underbrace{12}_{a}\;\underbrace{34}_{b} \qquad Y = \underbrace{56}_{c}\;\underbrace{78}_{d}$$

a and c represent the first n/2 digits of X and Y, Similarly b and d represent the last n/2 digits of x and y.

This algorithm involves 4 main steps

Step 1: compute $a \times c = 12 \times 56 = 672$ —— ①

Step 2: compute $b \times d = 34 \times 78 = 2652$ —— ②

~~Step 3: compute (a+c) x (b+d)~~

Step 3: compute $(a+b) \times (c+d) = 46 \times 134 = 6164$ —— ③

Step 4: compute ③ − ② − ① = 6164 − 2652 − 672 = 2840

Finally, multiply the o/p of step 1 by 10^n the output of step 4 by $10^{(n/2)}$ & add them both with the o/p of step 2

$6720000 + 284000 + 2652 = 7006652$

Answer is :- 7006652

The time complexity would be : $T(n) = \theta(n^{\log_2 3})$

$\log_2 3$ is approx equal to $1.585 < 2$

It will perform better.

Q3

	Dynamic	Greedy
1)	It has a bottom-up approach that builds-up solution by solving sub problems	It makes locally optically choice at each step with hope of finding globally optimal
2)	DP stores the solution to sub problems & then reuses them when needed	It doesnt consider the future consequences of current choice.
3)	Slower & complex	Faster & simpler
4)	Always provides a optimal solution	It may or may not provide a optimal solution

Q4 Explain all pair shortest path algorithm. Apply Floyd's algorithm for the example

→ It follows the dynamic programming approach to find the shortest path

→ Algorithm

n = no. of vertices

A = matrix $n \times n$

for $k=1$ to n

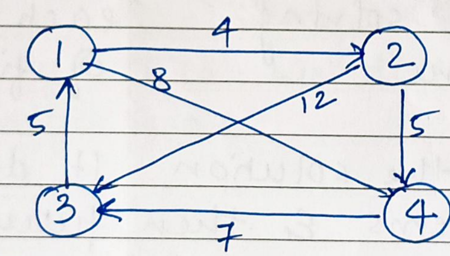
  for $i=1$ to n

    for $j=1$ to n

$$A^k[i,j] = \min \begin{cases} A^{k-1}[i,j] \\ A^{k-1}[i,k] + A^{k-1}[k,j] \end{cases}$$

return A.

→



D_0 =

d

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & \infty & 8 \\ 2 & \infty & 0 & 12 & 5 \\ 3 & 5. & \infty & 0 & \infty \\ 4 & \infty & \infty & 7 & 0 \end{array}$$

π

$$\begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & - & 1 & - & 1 \\ 2 & - & - & 2 & 2 \\ 3 & 3 & - & - & - \\ 4 & - & - & 4 & - \end{array}$$

$$D_1 = \quad d \qquad\qquad \Pi$$

$$\begin{array}{c|cccc|}
 & 1 & 2 & 3 & 4 \\
\hline
1 & 0 & 4 & \infty & 8 \\
2 & \infty & 0 & 12 & 5 \\
3 & 5 & 9 & 0 & 13 \\
4 & \infty & \infty & 7 & 0
\end{array}
\qquad
\begin{bmatrix}
- & 1 & - & 1 \\
- & - & 2 & 2 \\
3 & 1 & - & 1 \\
- & - & 4 & -
\end{bmatrix}$$

$(k=1)$

$$D'(2,3) = \min \begin{cases} D^0(2,3) \\ D^0(2,1) + D^0(1,3) \end{cases}$$

$$= \min \begin{cases} 12 \\ \infty + \infty \end{cases}$$

$$= 12$$

$$D'(2,4) = \min \begin{cases} D^0(2,4) \\ D^0(2,1) + D^0(1,4) \end{cases}$$

$$= \min \begin{cases} 5 \\ \infty + 8 \end{cases}$$

$$= 5$$

$$D'(3,2) = \min \begin{cases} D^0(3,2) \\ D^0(3,1) + D^0(1,2) \end{cases}$$

$$= \min \begin{cases} \infty \\ 5 + 4 \end{cases}$$

$$= \cancel{\min}\ 9$$

$$D'(3,4) = \min \begin{cases} D^0(3,4) \\ D^0(3,1) + D^0(1,4) \end{cases}$$

$$= \min \begin{cases} \infty \\ 5 + 8 \end{cases}$$

$$= 13$$

$$D^1(4,2) = \min \begin{cases} D^0(4,2) \\ D^0(4,1) + D^0(1,2) \end{cases}$$

$$= \min \begin{cases} \infty \\ \infty + 4 \end{cases}$$

$$= \infty$$

$$D^1(4,3) = \min \begin{cases} D^0(4,3) \\ D^0(4,1) + D^0(1,3) \end{cases}$$

$$= \min \begin{cases} 7 \\ \infty + \infty \end{cases}$$

$$= 7$$

$$D^2 = \quad d: \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 16 & 8 \\ 2 & \infty & 0 & 12 & 5 \\ 3 & 5 & 9 & 0 & 13 \\ 4 & \infty & \infty & 7 & 0 \end{array} \qquad \pi: \begin{bmatrix} - & 1 & 2 & 1 \\ - & - & 2 & 2 \\ 3 & 1 & - & 1 \\ - & - & 4 & - \end{bmatrix}$$

$$D^2(1,3) = \min \begin{cases} \infty \\ 4+12 \end{cases} = 16$$

$$D^2(1,4) = \min \begin{cases} 8 \\ 4+5 \end{cases} = 8$$

$$D^2(3,1) = \min \begin{cases} 5 \\ 9+\infty \end{cases} = 5$$

$$D^2(3,4) = \min \begin{cases} 13 \\ 9+5 \end{cases} = 13$$

$$D^2(4,3) = \min \begin{cases} 7 \\ \infty+12 \end{cases} = 7$$

$D_3 =$

$$d \quad \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 16 & 8 \\ 2 & 17 & 0 & 12 & 5 \\ 3 & 5 & 9 & 0 & 13 \\ 4 & 12 & 16 & 7 & 0 \end{array} \qquad \Pi \quad \begin{bmatrix} - & 1 & 2 & 1 \\ 3 & - & 2 & 2 \\ 3 & 1 & - & 1 \\ 3 & 3 & 4 & - \end{bmatrix}$$

$$D^3(1,2) = \min\begin{cases} 4 \\ 16+9 \end{cases} = 4$$

$$D^3(1,4) = \min\begin{cases} 8 \\ 16+13 \end{cases} = 8$$

$$D^3(2,1) = \min\begin{cases} \infty \\ 12+5 \end{cases} = 17$$

$$D^3(2,4) = \min\begin{cases} 5 \\ 12+13 \end{cases} = 5$$

$$D^3(4,1) = \min\begin{cases} \infty \\ 7+5 \end{cases} = 12$$

$$D^3(4,2) = \min\begin{cases} \infty \\ 7+9 \end{cases} = 16$$

$D_4 =$

$$d \quad \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 15 & 8 \\ 2 & 17 & 0 & 12 & 5 \\ 3 & 5 & 9 & 0 & 13 \\ 4 & 12 & 16 & 7 & 0 \end{array} \qquad \Pi \quad \begin{bmatrix} - & 1 & 4 & 1 \\ 3 & - & 2 & 2 \\ 3 & 1 & - & 1 \\ 3 & 3 & 4 & - \end{bmatrix}$$

$$D^4(1,2) = \min\begin{cases} 4 \\ 8+16 \end{cases} = 4$$

$$D^4(1,3) = \min\begin{cases} 16 \\ 8+7 \end{cases} = 15$$

$$D^4(2,1) = \min\begin{cases} 17 \\ 5+12 \end{cases} = 17$$

$$D^4(2,3) = \min\begin{cases} 12 \\ 5+7 \end{cases} = 12$$

$$D^4(3,1) = \min\begin{cases} 5 \\ 13+12 \end{cases} = 5$$

$$D^4(3,2) = \min\begin{cases} 9 \\ 13+16 \end{cases} = 9$$

Final :-

$$d: \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & 0 & 4 & 15 & 8 \\ 2 & 17 & 0 & 12 & 5 \\ 3 & 5 & 9 & 0 & 13 \\ 4 & 12 & 16 & 7 & 0 \end{array} \qquad \Pi: \begin{array}{c|cccc} & 1 & 2 & 3 & 4 \\ \hline 1 & - & 1 & 4 & 1 \\ 2 & 3 & - & 2 & 2 \\ 3 & 3 & 1 & - & 1 \\ 4 & 3 & 3 & 4 & - \end{array}$$

If we need path

Eg 1: 4 to 2

$$\Rightarrow 4 \xrightarrow{7} 3 \xrightarrow{5} 1 \xrightarrow{4} 2$$

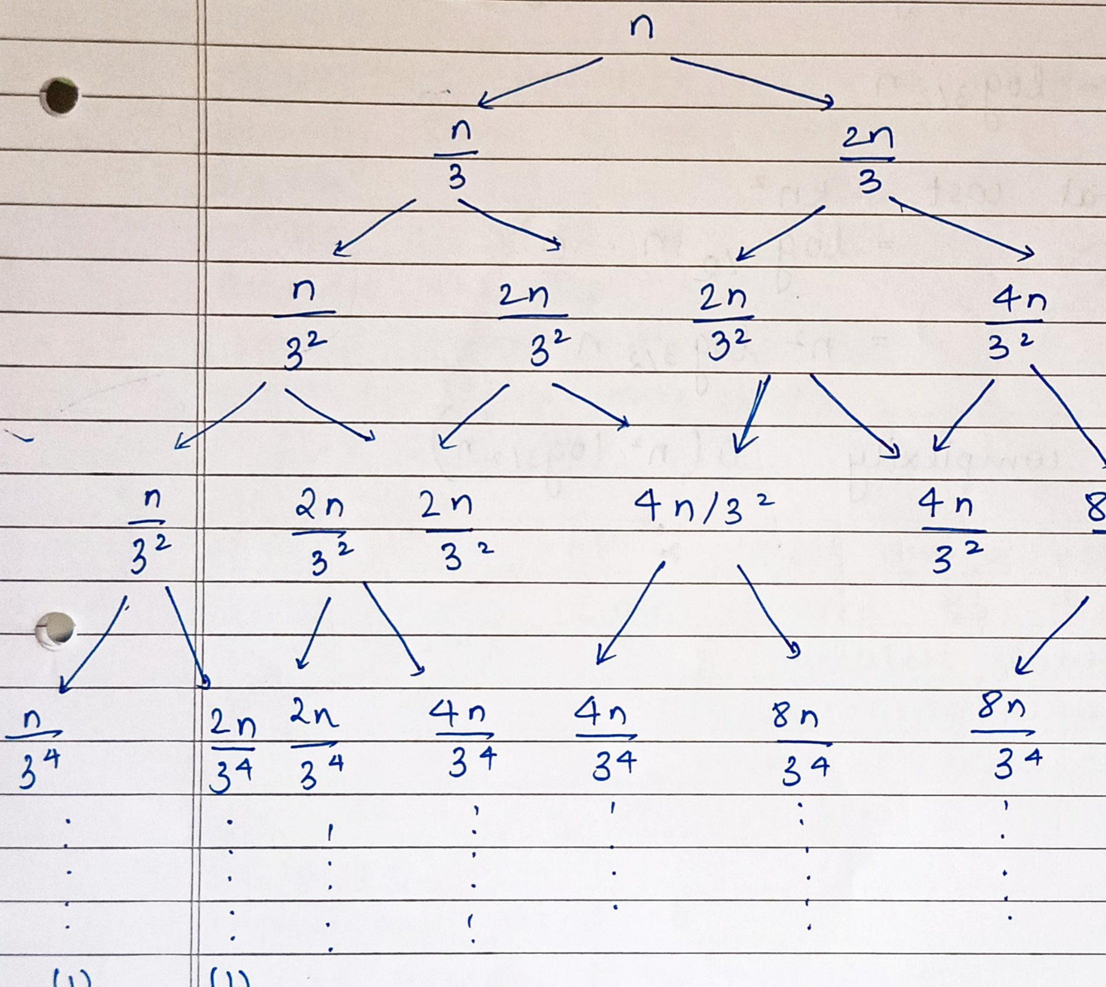
Eg: 3 to 4

$$\Rightarrow 3 \xrightarrow{\cancel{3}\,5} 1 \xrightarrow{8} 4$$

Q5 Find the time complexity of the following reccurrence relationship using reccurrence tree method.

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

→



$\Rightarrow n^2$

$\Rightarrow \frac{n^2}{3^2} + \frac{4n^2}{3^2} = \frac{5n^2}{3^2}$

The cost of each corresponding division of the at each level is n^2

$$\left(\frac{2}{3}\right)^k n = 1$$

$$k \log\left(\frac{2}{3}\right) + \log n = 0$$

$$\therefore k = \frac{-\log n}{\log(2/3)} = \frac{\log n}{\log(3/2)} = \log_{3/2} n$$

$$\therefore k = \log_{3/2} n$$

$\therefore$ Total cost $= kn^2$

$$= \log_{3/2} n \cdot n^2$$

$$= n^2 \log_{3/2} n$$

Time complexity $O(n^2 \log_{3/2} n)$