Design And Analysis of Algorithm (DAA)

Tutorial -2

ļ				
	1. void fun (int n) {	j=	1=0	No. of
	int j=1, 1=0;	1		-
	while (ikn) {	2	3	Hing
	1=1+11	3	6	Twen
	j++ }}	4	10:	MON.
	4			

Time tomplanity will be O(2")

$$f(n-1) \qquad f(n-2) \qquad f$$

Time Complexity = 0(2n)

3.i) n/lagn)

(ii) n³

for (izo; ixn; i+t)

for (jzo; jxn; j+t)

for (kzo; kxn; k+t)

(iii) log (log n)

int func (int n)

if (n==1)

return n;

else

return func (Jn) + func (Jn);

Aug 4. $T(n) = T(\frac{n}{2}) + T(\frac{n}{2}) + (n^2)$

Using master's method; a=2 b=2 c=1 $f(n) > n^2$ $f(n^2) > 1$ Time Complexity = $D(n^2)$

Aug 5. int func (int n) {.

for (int iz); ik=n; i++) \(\)
for (int j=1; j\x n; j\x =1) \(\)

771

Time Complexity = O(nJn)

Au 6. for (int i=2; i=n; i=pow(i,k))

C++ !

where Is is a constant.

Ans 7. T(n) = T (99 n) + T (n)

 $f\left(\frac{99}{100}\right) \qquad f\left(\frac{n}{100}\right)$

Time Complexity = O(log n)

Aug 8.

a) $100 \text{ K} \log (\log (n)) \leq \log (n) \leq 5n \leq n \leq \log (n) \leq n \log (n)$ $\leq n^2 \leq 2n \leq 2^{2n} \leq 4n \leq n!$

b) $1 < \log(\log(\ln)) < \sqrt{\log(2^n)} < \log(\ln) < 2(\log(\ln)) < \log(\ln) < 2(2^n) < n$ $< \log(\ln) < n < 2n < 4n < n^2 < \log(2^n) < 2(2^n) < n$

c) $96 \times \log_2(n) \times \log_3(n) \times \log_3(n) \times \log_3(n) \times \log_2(n) \times \log_2(n) \times \log_3(n) \times \log_$

Name - Sristi Gangwar Section - E

Roll no. - 2014897 (56)