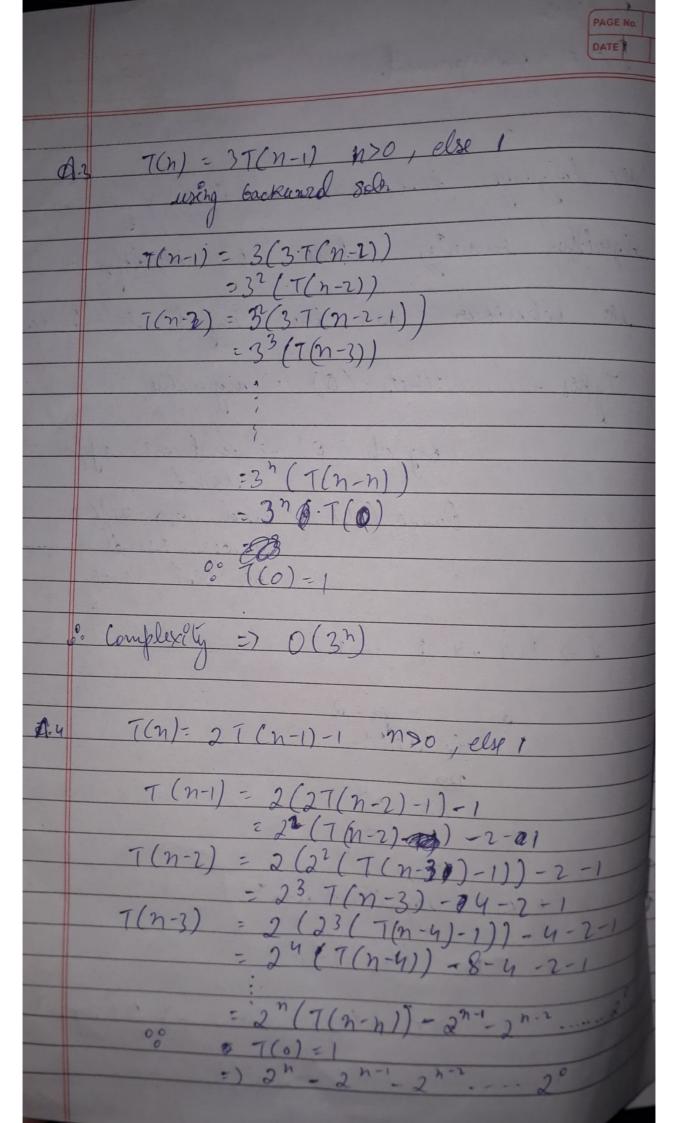
Rakshit B/39 DAA - assignment. At De languages that allow us to analyse an algorethen running time by identifying els behaviour as the Enfait seg increases Types - Eneta (a) -) ang value Big oh (O) - worst cast, eigher bound. Duega (30) -) used to define 1 =) 1, 2, 4, 8 - n Klog2 = leg 2(n) + log22 K= log_(n) 41 => 0 ((eg i(n) +1) =) 0 (leg n)



= 2h - (2h-1) à. Qu complexity => O(1) VI THE 5: Complexity = D(Jn) Complexity = 0 (In A.7. leops=) i Lougherity of n x logh x logh =) 0 (n (lag2n)2

Juternost loop Complexity -) £1.9 n/3 fines complexity = 0 (n logn) Since polynomials grow slower, exponentials. @? has an asymptomatic upper bound of o (an) for, a:2, no=2

PAGE No. DATE A. u K(X+1) = h ... 3. Complexity = Nn , T(1) 0 =0 T(0)=0 A-12 T(n)= T(n-1)+I(n-2)+1 n>1 Let (n-1) ~ 7(n-2) T(n)= 2 T(n+1)+1 using backward solv T(n) - 22(T(n-2)+1)+ = '4 (T(n2))+3 · To 7(n-1)= 2* 7(n-3)+1

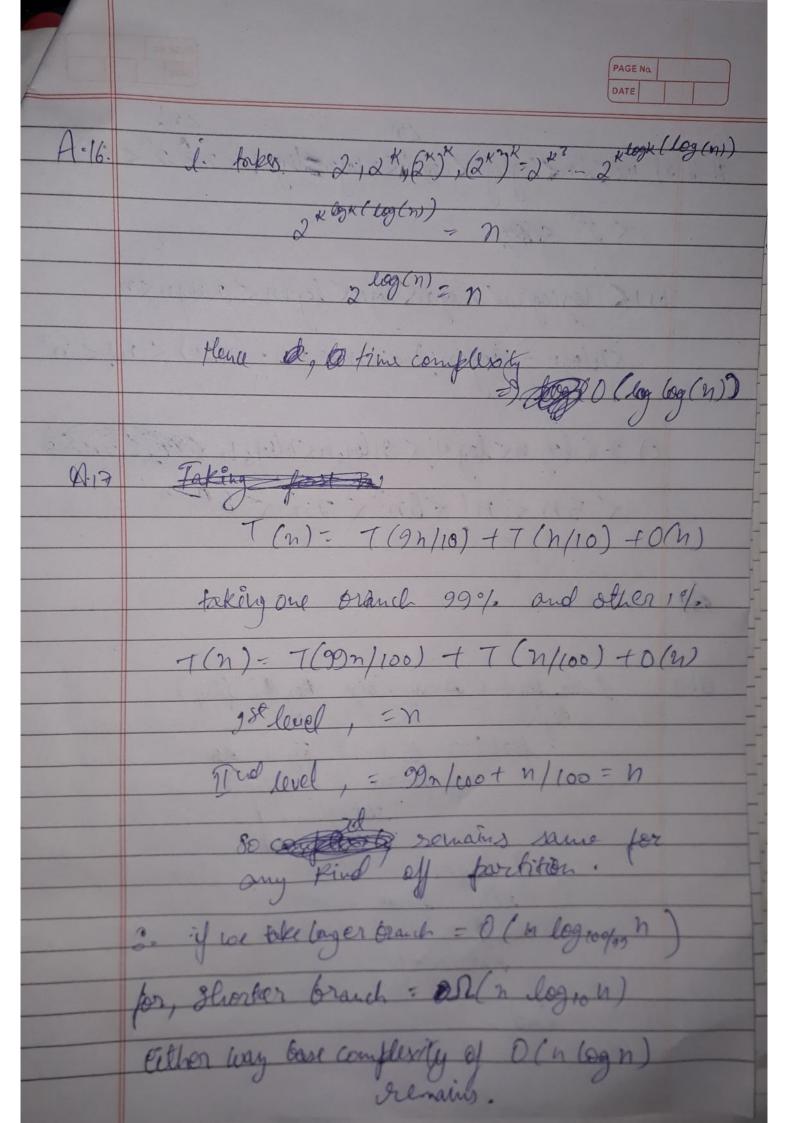
void fun (intr) ? for (i=1 to n) {

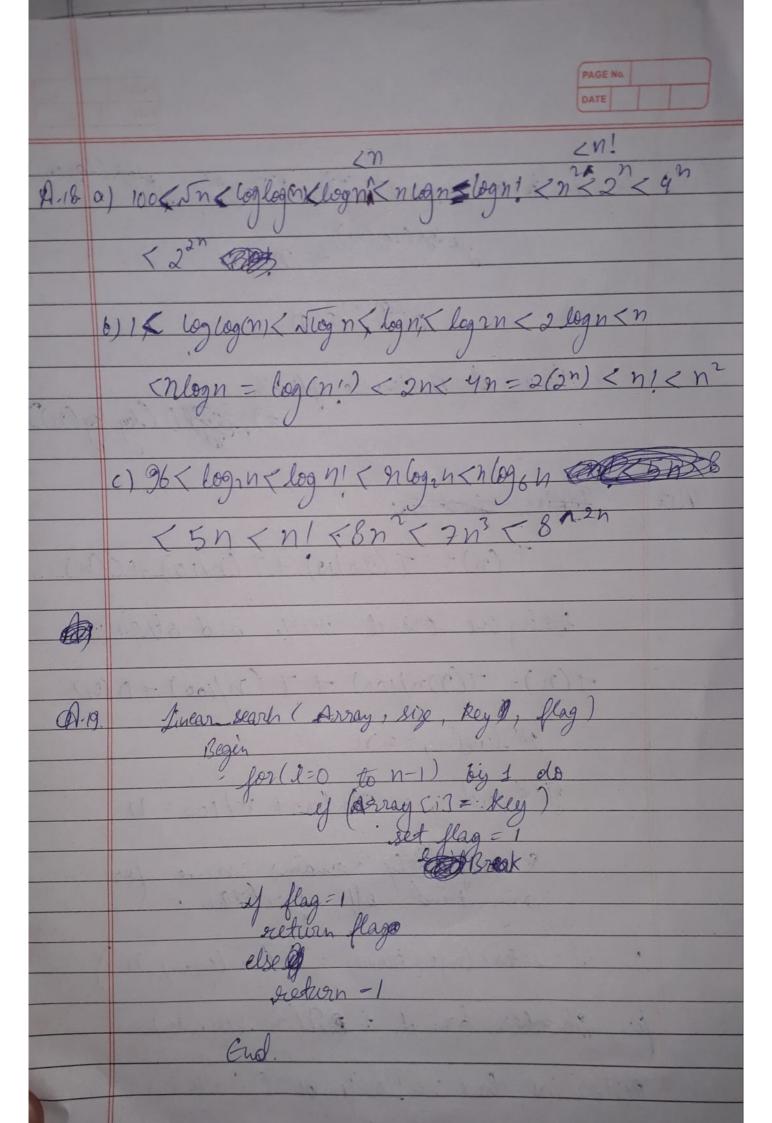
for (j=1 to n) {

for (k=1 to n)

// Some O(1) fast. log log (n) Void func (int n) [for (i:n; i > 1; i= fow (i, K)) 1/50 me o(1) struct

	PAGE NO.
A-10	T(n) = T(On/4) + T(n/2) + cn2
	1990 assume ((n/2) = 17 (n/9)
	T(n)=2T(n/2)+(n²
	= log 2 -1
	Complexity D(n2)
A-5	2 Atmes
	n/2 dimes
	n/n times lega
	é. Confailig Don logn





Recursal Herative 1.20 insertion (int a [], int n) insertion (a, iti, n Decause online algo sloesnot know the verole Worst 8 (n2) 0 (22) n (san) 9 (nean) 2 (nlogn) O (hlogn) O (nlogn) (n (ogn)