	Name-Prajjuwal Mehta Sec - D Rollno - 27 Page No
	Dota Analysis of Algorithms
01	No Portion Responds to the first
· 1 /2	These notations lave used to tell the
	complexity of an algorithm when
	"nout is very large."
(1	DO 5 200 1 200 0 10
1	Big-OCO)
	CP trokano moras.
	f(n) = o(cg(n))
	iff (a) O Maring
1	A(m) e cog(n)
	Thomas of the
	some constant < >0
2	Big Imega (1)
	+(n) = 2 (a(n))
	g (m) is "tight" Louver bound
0	V 2000
	+(n)= -2 (g(n))
	iffer in a control of the second
-	T(n) >(g(n)
/ 1	The molecule
	some constant C>0
	(100 9 0) x (10) 10 to 10 to 10
	The Marine Control of

Theta (O) t(n) = 0 (g(n)) g(n) is both "tight" upper & low bound of t(n). +(n)= 0 (g(n)). iff $c_1 g(n) \leq f(n) \leq c_2 g(n)$ $+ n \geq \max(n_1, n_2)$ $+ some constant c_1 > 0$ Small 0 (6) 4 f(n) = 0 (g(n)) g(n) is suppose bound of the function f(n) = 0 (g(n))+(n) ~ (. g(n) V n z no and + constant, c>0 Small omega ((w) 5 g(n) is hower bound of func for +(n) = (w(g(n)) when + (n) > c. g (n) + n>no and + c>o

02 tor (= 1 ton) // i=1, 2, 4, 8, ... €) i = i * 2 } 110(i) => 5 1+2+4+8+ · + · +n Kt town of GP => TK = agk-1 1) 11 -11 thing n= 1*ak-1 1) / 5) (5 00) / n= 2k-1 1) FS = (M) II 2 log 2n = Klog 2 log 2n = K $K = \log_2 2 + \log_2 n$ $K = 1 + \log_2 n$ May OC log 20)

03 T(n) = {3T(n-1) if n>0, otherwing T(n) = 3T(n-1) - (1)but n = n-1 in (1) T(n-1) = 3T(n-2) - (2) Fut T(n-1) in (1) T(n)=3(3T(n-2)) T(n) = q T(n-2) - (3)put n= n-2 in (1) T(n-2) = 3T(n-3) - (4)1 2 Amaput in (3) T(n) = 27 T(n-3) - (5)T(n) = 13k T(n-k) EXPORTED NO (O) 1 (3) 1 (1) K=1 T(n)= 3M-1-T(n=n+1) let k=n T(n) = 3 n T(0) $T(n) = 3^n$ Complexity = 0 (3")

QY T(n) = \$2T(n-1)-1 if n>0, otherwise (a 8/2) 16 By * T(n) = 2T(n-1) - 1 (1) put n=h-11. in (1) T(n-1)=2T(n-2)-1-6) put (T(n-1) from (2) to (1) T(n)=27 (2T(n-2)-1)-1 T(n) = 4T(n-2) - 3 - (3)put n = n-2 in (1) T(n-2) = 2T(n-3) - 1 - (4)but T(n-2) from (4) to (3) T(n) = 4(2T(n-3)-1)-3T(n) = 8T(n-3)-7 - (5) (++++//T(n) = 2KT(n-K)-2K-1-(6) Let n-k=1 $T(n)=2^{k}T(1)-2^{k-1}$ $T(n) = O(2^{k}) = O(2^{n})$ T(n) = 2 5 T(1) - 2 1-1-1 TG)= 200 20 -1 Ten) = Oar)

05 int i=1, b=1; while (sa<=n)) 1011 5 Hard Hard Mary 1981 &=&+i; printf (" ##"); 3 No 18 1 10 21 1 20 21 1 = 02, 3, 14, 15, 6 ... K D= 3, 6, 10, 15, XK at then doop not n= < & nature Ka iteration -> 2+2+3+4+ ... + K=n = 1+ (K+(K+1)) 12 = n $K^2 = n$ K= In = O(In)

06 word function (int n) 1 bor 10 mint i, count = 0; Hor (i=1: i*i<=n: i++) (40) 10 (11 count ++. (oblight n / (To not 1-12- (i) = 1,2,3,41.21. Ja 5 1+2+3:- 10+ To TChYE To x (Jn+1) (n)0 = cn)T

07 void function (int n) 11/10 int e i, j, k, count = 0; for (1=112; 1=10; 1++) - oderga) 1 for (j=1,j<=n;j=j*2)-dago tor (K=1; KZ=n; K= K*2) count ++ O (nlog2n) function (int) if (n==1) return; for (i=1 ton) for (j=1 ton) printf (" *"); function (n-3);

void function (int n) Od jor (i=1.ton) tor(j=1: j<=n; j=j+i) Brint ("*"); i=1, j=1,3,3,4,5,6,.... i= 3, i= 1,47,10,13 · - - - 013 T(n)= n+ n+ n+ n $T(n) = n(t_1 + t_1 + t_2 + t_3 + t_4 + t$ = n logn O(n logn)

010 (a lib) parting book as given nk & cn relation obstavein no & co is nk = 0 (ch) itie ous inkle on V holing & rome constant for no=1 C,=2 => 1K < 0.2' => no=14 C=2