Jonathan Asprilla Saavedra - 8926036 Tarea 1 2. Clasificar por orden asintótico las siguientes funciones >log(Vn) < clog(n!) Aplicando evler 2/8g(Vn1) L & c. log(n!) Vn1 L c.n! Para todo nz no n≥2; √2 < 21 > log(n!) < cn2logn Aplicando euler €log(n!) = ecnilogn n! < cn3 n=1, c=1 Para todo n ≥ no n ≥ 1; 1141 n2,5 < n2 logn. c Es falso porque C= 4 12,5 £ 12 log 1.1 tonzlogn Ecn35 Para todo n21

	minit sentites in barbarable in the sail in supposed in	-
1 5	mil scholos de de la	-
2400000	Talson posquel of tomas inspections as many many	-
	N= 2 (=1	
	√51. ≤ 5 ^{2,5}	-
4	120 4 55,9	=
2	: n2,5 4 cn! + n 25	•
	$\Rightarrow 2^n \leq cn!$	
	Falso, porque	
	n=1, c=1	
	$2^2 \leq 1!$	
	$4 \le 1$ $1 \le 1 \le 1$ $1 \le 1 \le 1$	(L)
	:. n! \(\c2 \) \\ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \ \	
	Ordenado ascendentemente, seña:	
1	Otagriga oct	
	1. log(1n)	
	2. Log(n!)	
	3. n²log(n) 4. n²,s	
	4. n ^{2,5}	
	5. nl	
	6. 22"	
	2 TED ± G KY	
		-
		*

2.2.12 a) Sife O(n), entonces f2 & O(n2) (o) fe ocn) Se tienen valores no ENy CE Roo | 4 n ≥ no hands se complagre f = c.n f = cn n=1, c=2 f ≤ 2n; + n≥no, n≥1 :. Para f2 E O(n2) f2 < cn2 con no=1 y c=2 :+ oping f2 6 2n2 4n | n ≥1 b) Si f & O(n), entonces 2f & O(2n) f 6 0(n) Se tienen Valores no EIN A CERSO | 4 n = no Se compla que f = c.n f = (n no=1, C=2 f=2n; + n=no, n=1 · . Para 2 f & 0(2") 2f < C. 2n, no=1, C=2 log(2+) ≤ log(2")·c flog(2) < nloy(2).c flog(2) < 2 n log(2) $f \leq 2n$.. f ≤ 2n +n≥no 1 n≥1

	Punto 3:
	Fjercicio 4:
	7 91 esta antes de 95, porque si se toman los
	loganitmos, se comparara logn con lognit logs
	logn + log (logn) > logn; ocumbiando de variable pera
	verlo mais sencillo sería z= logn; /2 = 21/2 contra
	all mas senano sena 2
	$z + \log z \ge z$.
	(1 - 1 ³ cmco - 5 5 55 55
	-> 95 es menor que 93, ya que (logn)3 crece más rápido
	que logn. Ambos son polinomios en logn pero (logn)3
	tiene mayor grado.
	- 93 está antes que que, porque si dividimos ambos entre n
	omparariames (log n)3 con n 1/3. Así pues, los logoritmos
	Execon mais lento que los exponenciales.
	-> gy está primero que gz, gracias a que los polinomios
	over a stable of the event city es
	crecen mas lento que los exponenciales.
_	20 15 1 0 0 0 100 0 0 1 1 max to compared to the
	-> 92 está antes que 9, yarque, al tomar los loganitmos,
	se compararia n con n2 y n2 es el polinomio con mayor
	90090
-9)	-> 97 está antes que 96 ya que, st comparamos nº to 2",
	los polinomios crecen mas lento que los exponenciales

ordenados:

91) 2 logn 95) nlogn 93) n(logn)³ 94) n4/3 92) 2ⁿ 94) 2^{n²} 96) 2ⁿ

2.3.1 del Punto 2 de la Tarrea

a) $\Omega(f) = \Omega(cf)$ - Tomado de las notas de clase Si g: MI -> R zo es dal que g $\in \Omega(f)$, basta con elemostrar g $\in \Omega(cf)$. Si g $\in \Omega(f)$, entances hay No. \in Ny Co \in Rxo | g(n) \geq Cof para news forme $\Omega = ho$ y G = Co, y note que para nen se tiene: $g(n) \geq Cof(n)$ (por suposición) = C(cf(n)) (por definición de G)

luego, g & M(cf) con destigos n. y CI

Si g: $N + R \ge 0$ es tal que g $\in \Omega(cf)$, basta con demostrar $g \in \Omega(f)$. Si $g \in \Omega(cf)$, entonces hay no $\in \mathbb{N}$ y $\in \mathbb{R}$ tales que $g(n) \ge \operatorname{Co} cf$ para $n \ge n_0$. Tome $n_1 = n_0$ y $ext{C}_1 = \operatorname{Co} ext{C}_1$ y note que para $n \ge n_1$ se tiene: $g(n) \ge \operatorname{Co} cf(n)$ $= \operatorname{C}_1 f(n)$

luego, g E-12(f) con testigos na y C1.

 $\rho(t) = \rho(ct)$

Si g: N > R > 0 es tal que gleno (f), basta con o 2 demostrar g & O (cf). Si g & O (f), entonces hay no EN y Co E R > 0 | g (n) & Cof para n > No - Tome n, = No y C = Co, y note que para n > n, se tiene:

	zohorsko
	$q(n) \leq cof(n)$
	$= C_1 cf(n)$ (28)
	Luego, g & D(cf) con testigos ni y cris (E)
	Principal Care Care Care Care Care Care Care Care
	· ≤: g: N → R≥0 es tal que g € θ(cf), basta con
	demostrar q & O(f), Si q & O(cf), entonces hay
	ha EN y co ER so tales que q cn) & co cf para n ≥ no.
0 % 3	tome no = no V C1 = COC. V note que para nani se tiene.
	9(n) = Cocfant of ab & chart 1sh L.C.20
	a) I (f) = I (cf)
	Luego, g & O(f) con testigos ni y ci.
	The second of th

Punto 4.

Exercicio 13: Inversions

Let A[1...n] be an array of n distinct numbers. If i < j and A[i] > A[j], then the pair (i,j) is called an inversion of A.

We need to recursively divide the array into halves and count number of inversions in the sub-arrays. This will result in logn steps and $\theta(n)$ operations in each step to count the inversions. All in all a $\theta(n\log n)$ algorithm

def Inversions (A, P, r)

it b > L

return 0

9 = [(p+r)/2]

left = Inversions (A, p, q)

right = Inversions (A, 9+1, r)

inversions = left + right + merge (A, p, q, r)

return inversions

Modified merge - sort

+ ofnis Modified merge -sort det Merge (A, p, p, r): proissont is aciding h1 = 9 - p + 1no= v-9 let [[1.. n] and P[1.. nz] be new arrays for i = 1 to n1 [1-i+9] A = [1] for j=1 to n2 1 to n2 R[i] = A[9+j][[n1+1]= 00 R[n2+1] = 00 1=1 i = 1inversions=0 for K=P tor if L[i] = R[j] A[K] = I[i] i= i+1 else: inversions = inversions + (n1-i+1) A[K]=R[j] i = i + 1return inversions

Ejercicio 31: Fixed Point

Suppose we are given an array A[1..n] of n distinct integers, with could be positive, negative, or zero, sorted in increasing order so that A[1] < A[2] < ... < A[n].

a) Suppose we define a second array B[1..n] by setting B[i] = A[i] - i for all i. For every index i we have $B[i] = A[i] - i \leq (A[i+j]-1) - i = A[i+1] - (i+1) = B[i+1]$

A[i]=i if and only if B[i]=0

def Find Match (1, r):

if 1,> r:

return None

mid = (1+x)/2

if Almid = mid

B[mid] = 0

return mid

else if A[mid] < mid

B[mid] < 0

return FindMatch (mid+1, r)

98/9

B[mid] > 0

return Find Match (1, mid-1)

- 3 ctmg b. der Find Match Ros (A[1..n]): +md bexif 18 oppress if A[17=1 Suppose we are given an array ALL. In I grant Stistinct integers, with carle of positive regative or sero, sorted in increasely order so that A[1] < A[2] < ... < A[n]. . gnow muter Again, the array B[1...n] defined by Setting B[i]=A[i]=i is Sorted in inageosing order. It follows that if A[1] >1 (that is, B[1] >0), then A[i]>i (that is, B[i]>0) for every index 2. A[1] cannot be less than 1.