

## Parcial-1-2018-RESUELTO.pdf



un\_saltador



Análisis y diseño de algoritmos



2º Grado en Ingeniería Informática



Escuela Técnica Superior de Ingeniería Informática Universidad de Málaga



La mejor escuela de negocios en energía, sostenibilidad y medio ambiente de España.

**Formamos** talento para un futuro Sostenible



2 100% Empleabilidad



Modalidad: Presencial u online



Programa de Becas, Bonificaciones y Descuentos



### ELIMINA IMPERFECCIONES RÁPIDA ABSORCIÓN NO DEJA RESIDUOS

## Todo lo que tu piel necesita cuando menos te lo esperas.



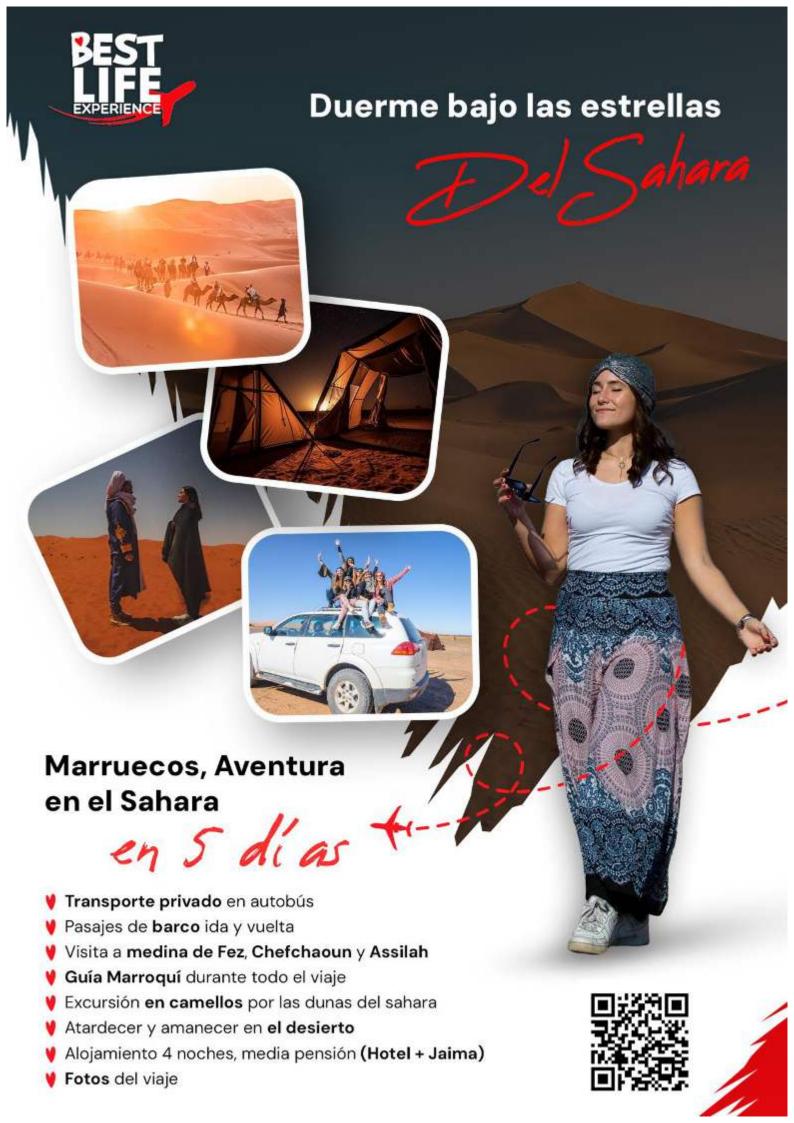


Análisis y Diseño de Algoritmos	Control Bioque 1
Alumno:	Grado:
Ejercicio 1	
Especificar un programa que dado un vector de enteros $v$ , e un vector consiste en sustituir en el vector todas las aparicies in modificar el resto de posiciones.	
Ejercicio 2	
Dada la ecuación en recurrencia $T(n)=2T(n-1)+n$ polinomio característico y dar su complejidad.	, con $T(0) = 0$ , se pide resolver por el
Ejercicio 3	
Se dispone de un array con el peso de n bolas, donde n que el resto y se desea encontrarla, pero no podemos acce directa. Para ello, se dispone de una función que nos da puede utilizarse 2k veces. Diseñar e implementar un algor tarea de encontrar esa bola. ¿Cuál es la complejidad del de recurrencia y utiliza el teorema maestro. Nota 1: Asumir que existe ya implementada una función vuelve la suma de los pesos de las bolas desde la posición Nota 2: Implementar el algoritmo con todo el nivel de de	der a la información del array de forma el peso de un grupo de bolas, pero solo itmo de Divide y Vencerás que realice la algoritmo diseñado? Plantea la ecuación a sum(int aff,int start, int end) que de- start a la posición end.
Ejercicio 4	
Marca la respuesta correcta introduciendo entre paréntesis	s (V)erdadero o (F)also.
<ol> <li>La clase NP podría resolverse en tiempo polinómio determinista ( )</li> </ol>	vo con un lenguaje de programación no
<ol> <li>Al transformar un problema de optimización a un pro en un factor log(n) ( )</li> </ol>	blema de decisión, la complejidad cambia.
<ol> <li>Los problemas NP-completos son los más complejos</li> </ol>	dentro de la clase NP ( )
4. Los problemas NP-hard son al menos tan complejos	como los NP-completos ( )
<ol> <li>El problema de la satisficibilidad se resuelve con alq y O(n) de complejidad espacial ( )</li> </ol>	goritmos $O(2^n)$ de complejidad temperal
<ol> <li>En la cluse espacial L el espacio es reutilizable, el espacio no es reutilizable ( )</li> </ol>	tiempo no. En la clase espacial EXP el
Puntuaciones: El: 6 pintos; E2: 7 pintos; E3: 14 pinto	se; E4: 3 puntos [1/2 punto acierto, -1/4

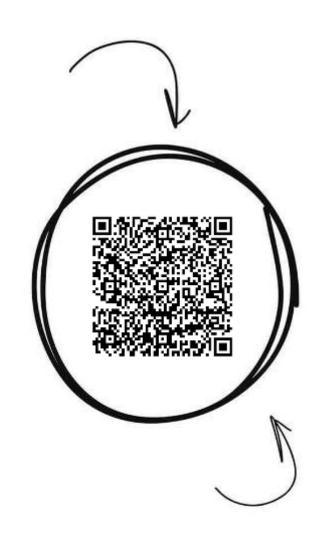


Control Cloque I wiso 2018 - 2019 (1) P = { l=v. length 20, vant = v }. positivizar (int []v) : void; a={ Vie [o, l) / vant [i] =0 = v[i] =0 1 A VIE [0,1) / vont [i] 70 = Vant [i] = v[i]}. (2) T(n)= 2T(n-1)+n, T(0) =0. T(n)-2T(n-1)=n1" => (x-2)(x-1)2=0. = T(n)= x2 + B+ 8n. T(0)=0, T(1) = 2. T(0)+1=1, T(2)=2. T(1)+2=4.  $\Rightarrow \quad \alpha + \beta = 0$   $2 \times + \beta + V = 1$   $4 \times + \beta + 2V = 4$  $\begin{pmatrix} 1 & 1 & 0 & 0 \\ 2 & 1 & 1 & 1 \\ 4 & 1 & 2 & 4 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 1 \\ 0 & -3 & 2 & 4 \end{pmatrix} \sim \begin{pmatrix} 1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 1 \\ 0 & 0 & -1 & 1 \end{pmatrix}$ Y=-1, B=-2, K= 2 = T(n)=2·2°-2-n ∈ Θ(2°).





# Análisis y diseño de algoritmos



Banco de apuntes de la



## Comparte estos flyers en tu clase y consigue más dinero y recompensas

- Imprime esta hoja
- Recorta por la mitad
- Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes
- Llévate dinero por cada descarga de los documentos descargados a través de tu QR





(3) n=3". A lo sumo, 24 vent la fención sum. public static int per Bola (intl Iv, int iy intrip, devole suma) { if (suma == 0) suma = sum (v, ing, sup), if (sup == in) return sup; else if (in) +1== sup | return (v[in] < v[sup] ? inj : sup); int m= (inf + sup)/2 double sum parcial = sum (v. e.g., m);
if (sum Parcial < (suma & cm-ing+11/(sup-ing+11)) [ retira por Bola (v. inf. m. suma - (suma \* (sup-m)/sup-inf+1)) felse f rebita pos Bola (v. m+1, sup, suma - (suma + (m-inf+1)/cup-inf+1) Adarationes: · La primera llamada a pos Bola debe ser can lot significates parametros: post Bola (v, o, v. length - 1, o); · Tome la suma como double perque trabajo con fracciones, y si pierdo decimales podría faller. lemplyided. T(n1= T(n/2) +n+4, WEN es via constante.





## ELIMINA IMPERFECCIONES RÁPIDA ABSORCIÓN NO DEJA RESIDUOS

## Todo lo que **tu piel necesita** cuando menos te lo esperas.

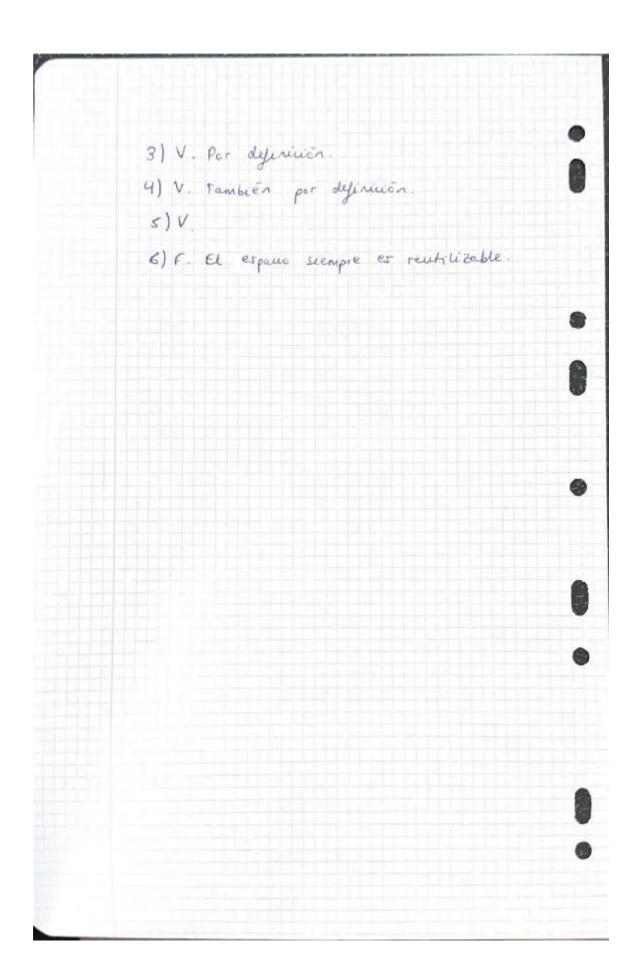


TU RUTINA ANTI -IMPERFECCIONES



-	log_1 = 0 < 1, dende el vitimo 1 es el esperiente
	de n, deducimos que nuestro algoritme es de orden n.
	Ademais, nuestro algoritmo sum se genta 1+ [log_2 tr] veces, puer 1+ [log_2 tr] = 1+ log_2 tr =
	= 24 Demostremos la utilina designaldad.
0	Sea f(x) = 24-log 24-1, f: (0,+00) - DNE.
	$g'(x) = 2 - \frac{1}{\ln 2} \cdot \frac{1}{W}$ , pero como $\frac{1}{\ln 2} \cdot \frac{1}{\ln $
	lugo VIIII, j'es estrictamente creuente, y como
•	geil=1, teremes que 1 = f(x) ∀x≥1, luege
	f(x17,0 Vx7,1 → 2x-log2x-170 Vx71 →
	= 2x > log 2 x +1, per lo que siempre vieremer
0	la función sum un número permitido de vecer.
•	(9) 1) Falso, no crister chapages de programación po deterministas. Cson palabras textuales del
	projesor, aunque yo pienso que es civito, pues de l'podría", y adjunto un texto al
	final de este POF con la explicación.
	2) No le sable el profesor, ni ye tempoco.
	Si alguno lo sabe, que me ovise por favor.
•	petinémice, pro ni idea n







## Complejidad computacional. Problemas P y NP

- Cualquier problema de optimización puede ser transformado en un problema de decisión. En el análisis de complejidad se manejan, por lo tanto, los problemas de decisión, que incluyen a los de optimización.
- Si se encuentra solución a un problema de decisión, entonces se encuentra también solución a un problema de optimización. Al estudiar los problemas de decisión, se pueden encontrar varias clases:
  - La clase de problemas P está formada por todos aquellos problemas de decisión para los cuales se tiene un algoritmo de solución que se ejecuta en tiempo polinomial en una máquina determinista.
  - La clase de problemas NP está formado por todos aquellos problemas de decisión para los cuales existe un algoritmo de solución que se ejecuta en tiempo polinomial en una (hipotética) máquina no determinista. Dicho de otro modo, no se ha encontrado un algoritmo determinista que lo resuelva en tiempo polinomial.



FPS 50+ CADA DIA

NO GRASO

Sabemos que la única luz que te da últimamente es la del flexo, pero está guay que para cuando acaben los parciales tu cara esté perfecta



TU RUTINA ANTI -IMPERFECCIONES

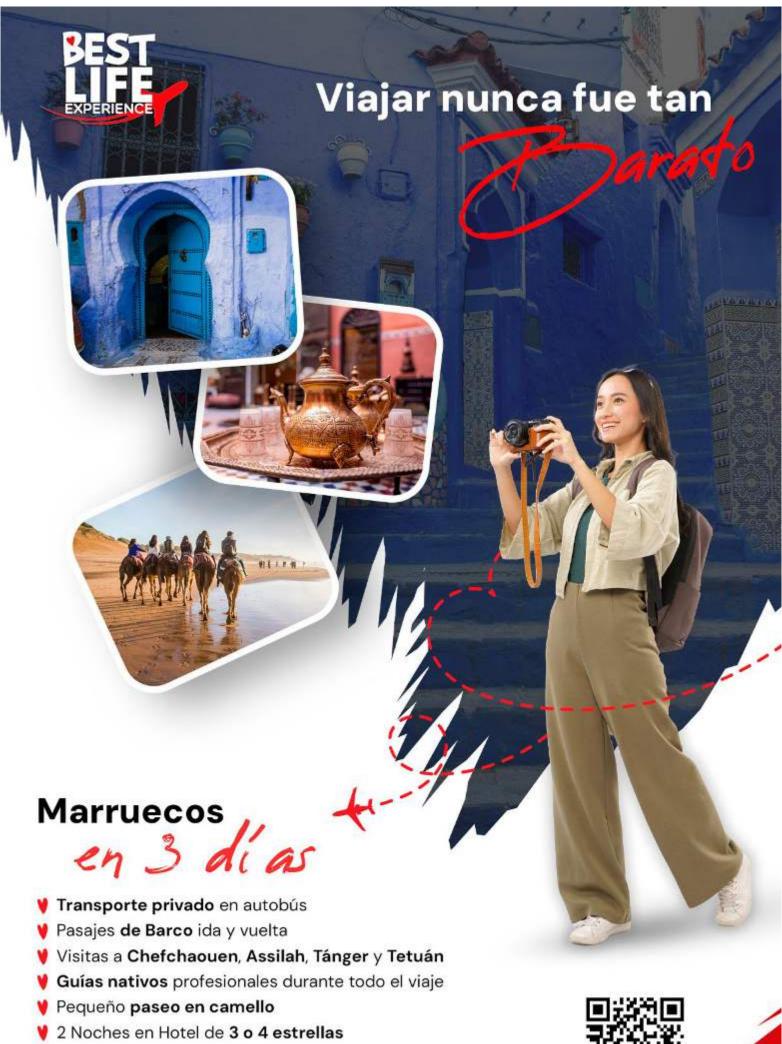


ARNIER PureActive

WUOLAH







♥ Desayunos y cenas incluidos
 ♥ Fotos del viaje



