

Apuntes-Tema-3.pdf



Juandf03



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.**

Más información
www.eoi.es

Formamos
talento para un futuro
Sostenible



100% Empleabilidad



Modalidad: Presencial u online



**Programa de Becas,
Bonificaciones y Descuentos**

Todavía estás a tiempo
de participar en el
sorteo y ganar...

**2 camisetas +
Pack Mahou**



→ Predicados equivalentes ($P \equiv Q$) si los satisfacen los mismos estados

6. Especificación de problemas

- Reforzar y debilitar un predicado

conjunción (\wedge)

disyunción (\vee)

- Convenios y notación

$$\{a..b\} = \begin{cases} \emptyset & \text{si } a > b \\ \{a\} \cup \{a+1..b\} & \text{si } a \leq b \end{cases}$$

$$N(P(x))$$

$$x \in \{a..b\}$$

→ n° de veces que
ocurre algo

Ej.

$$N(a[i] \neq 0) = \square$$

$$i \in \{0..a.length-1\}$$

Tema 3

"Dónde y Vencerás"

1. Introducción

Se divide el problema en instancias más pequeñas, las resolvemos
y combinamos las soluciones.

↓
¡RECURSIVIDAD!

$$T(n) = aT\left(\frac{n}{b}\right) + f(n)$$

① $\frac{n}{b}$ tamaño subproblema

② a n° subproblemas

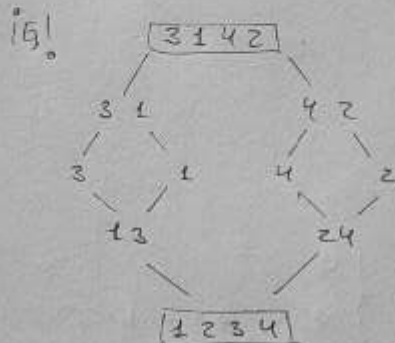
③ $f(n)$ complejidad combinar subproblemas
los soluciones

WUOLAH

2. Métodos de ordenación avanzados

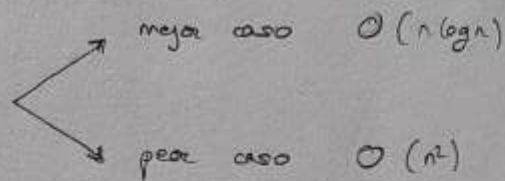
• Mergesort $O(n \log n)$

Algoritmo de ordenación por mezcla



• Quick Sort (Hoare)

Ordenación rápida



3. Búsqueda binaria recursiva

Encuentra la posición de un valor en un array ordenado.

Peor casos $\rightarrow O(\log n)$

4. Multiplicación de números grandes

① Multiplicación normal

ab, cd

$$ab = a \cdot 10 + b$$

$$cd = c \cdot 10 + d$$

$$ad \cdot 10 + bd$$

$$ac \cdot 10^2 + bc \cdot 10$$

$$\textcircled{ac} 10^2 + \textcircled{ad} + \textcircled{bc} 10 + \textcircled{bd}$$

$$(a+b) \cdot (c+d) = ac + ad + bc + bd$$

② Algoritmo

Reducimos el n° de productos

hacemos de otra forma

¿DÍA DE CLASES *infinitas?*



**masca
y fluye**



Análisis y diseño de algoritmos



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



Banco de apuntes de la

MUOLAH

1 Imprime esta hoja

2 Recorta por la mitad

3 Coloca en un lugar visible para que tus compis puedan escanar y acceder a apuntes

4 Llévate dinero por cada descarga de los documentos descargados a través de tu QR



IMPORTANTE

- Quiero Complejidad $\Theta(\log n)$ $T(n) = \Theta(n^d \log n)$

$$\boxed{a=1} \implies d = \log_b a = 0$$

$$T(n) = a T\left(\frac{n}{b}\right) + f(n)$$

$\implies \frac{n}{2}, \frac{n}{3}, \dots$

- Quiero complejidad $\Theta(n)$ $T(n) = \Theta(n^d)$

$$\boxed{d=1, \log_b a < 1}$$

$$\boxed{d < 1, \log_b a = 1}$$

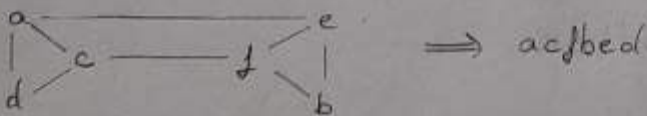
- Quiero complejidad $\Theta(n \log n)$ $T(n) = \Theta(n^d \log n)$

$$\boxed{d=1}, \log_b a = 1 \implies \boxed{a=b}$$

5. Recorridos en un grafo

$$\text{Grafo} \longrightarrow G(V, E) \begin{matrix} \xrightarrow{\text{aristas/aristas}} \\ \xrightarrow{\text{vértices}} \end{matrix}$$

• Recorrido en Profundidad (DFS)



Añado una arista en cada iteración

• Recorrido en Anchura (BFS)

acdfefb Añado todas las aristas

Aristas incidentes \longrightarrow nuevos vértices \longrightarrow aristas incidentes



¿DÍA DE
CLASES

infinitas?

Proceso para resolver Ejercicios Tipo Examen ▼

1. Especificación (pre, post)
2. Diseñar el algoritmo por Fuerza Bruta (dibujo + explicación + diseño del código)
3. Obtener la complejidad del algoritmo (T_{FB})
4. Diseñar un nuevo algoritmo aplicando Divide y Venceras (explicación en cada línea o método)
5. Obtener la complejidad del algoritmo (T_{Div})

$\frac{100\%}{100\%}$ bucles $\rightarrow \Sigma$
es recursiva $\rightarrow T_{Div}$ Maestría



masea
y fluye



WUOLAH