

Divide y Vencerás (Divide&Conquer)

Divide y Vencerás (Divide&Conquer)

Objetivos

- Comprender la finalidad de los algoritmos DYV.
- Comprender los elementos de los algoritmos DYV.
- Comprender el esquema de los algoritmos DYV.
- Saber calcular la eficiencia de los algoritmos DYV.

Divide y Vencerás (Divide&Conquer)

Comprender la finalidad de los algoritmos DYV

Los **algoritmos divide y vencerás** se aplican a problemas:

- Formados por **subproblemas disjuntos** (que no se solapan) y **que se pueden resolver de manera independiente.**
- Cuya solución final se puede expresar como **combinación** de las distintas **subsoluciones.**
- Cuyos subproblemas son **del mismo tipo** que el original (¡recursividad!).
- Cuyos subproblemas tengan aproximadamente el **mismo tamaño** (lleva la eficiencia a órdenes logarítmicos al dividir los problemas entre dos).

Siempre proporcionan la solución optima.

Divide y Vencerás (Divide&Conquer)

Comprender los elementos de los algoritmos DYV

- **Algoritmo específico:** Necesario para resolver subproblemas de tamaño muy pequeño.
- **Fase de división:** Descompone el problema en subproblemas.
- **Fase recursiva:** Soluciona recursivamente cada uno de los subproblemas.
- **Fase de combinación:** Construye la solución al problema original utilizando las soluciones a subproblemas de menor tamaño.

Divide y Vencerás (Divide&Conquer)

Comprender el esquema de los algoritmos DYV

Dado un **problema P**,

```
Función DV (P) {  
    if (P es sencillo) return específico(P);  
    else {  
        Descomponer P en subprob P1, P2, ..., Pa.  
        for (i=0; i<a; i++)  
            s_i=DV(P_i);  
        combinar s_* para obtener una solución s,  
        return s;  
    }  
}
```

Divide y Vencerás (Divide&Conquer)

Saber calcular la eficiencia de los algoritmos DYV

Si tardamos $T(1)$ en resolver un problema sencillo, $f(n)$ es una función no recurrente, y $n=b^x$:

$$T(n) = \begin{cases} T(1) & n = 1 \\ aT(n/b)+f(n) & n > 1 \end{cases}$$

Obsérvese que a y b dependen de en cuántos subproblemas se divide el problema en cada paso, y $f(n)$ depende del resto de operaciones (división y combinación).

Divide y Vencerás (Divide&Conquer)

Saber calcular la eficiencia de los algoritmos DYV

Si tardamos $T(1)$ en resolver un problema sencillo, $f(n)$ es una función no recurrente, y $n=b^x$:

$$T(n) = \begin{cases} T(1) & n = 1 \\ aT(n/b)+f(n) & n > 1 \end{cases}$$

Si tenemos:

$T(n) = aT(n/b)+cn^k$, con $T(1) = c$ (esto siempre ocurre)

El orden de eficiencia de estos algoritmos es:

$$T(n) \text{ es } \begin{cases} \Theta(n^k) & \text{si } a < b^k \\ \Theta(n^k \log_b n) & \text{si } a = b^k \\ \Theta(n^{\log_b a}) & \text{si } a > b^k \end{cases}$$

Divide y Vencerás (Divide&Conquer)

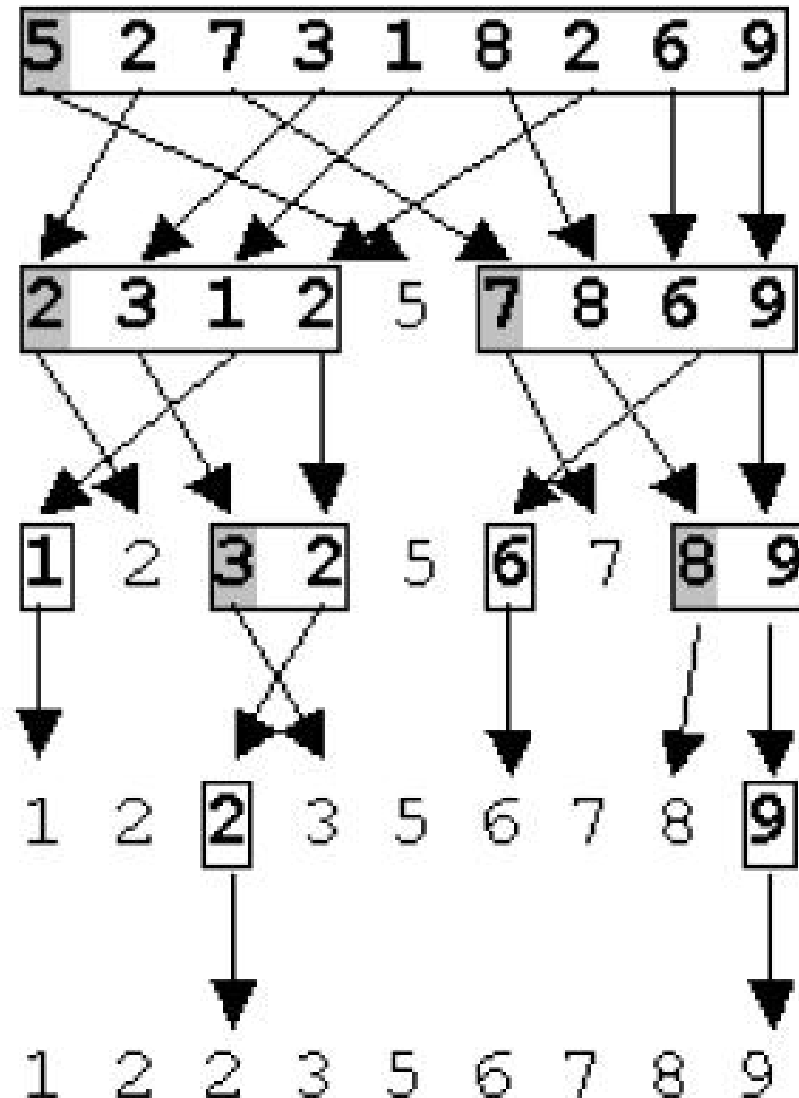
¿Qué hemos aprendido?

- **Finalidad de los algoritmos DYV.**
- **Elementos de los algoritmos DYV.**
- **Esquema de los algoritmos DYV.**
- **Calcular la eficiencia de los algoritmos DYV.**

Divide y Vencerás (Divide&Conquer)

Ejemplos

Ejemplo: Ordenación Quicksort.



Divide y Vencerás (Divide&Conquer)

Ejemplos

Ejemplo: Ordenación Mergesort

