### TPN°5: Divide & Conquer

Algoritmos y Estructuras de Datos II



## Decrease & Conquer

reducción en una constante El tamaño se reduce en un valor constante en cada iteración

variantes:

reducción por un factor constante

El tamaño se reduce en un factor constante en cada iteración

reducción de tamaño variable

El tamaño de la reducción del problema varia entre una iteración y otra.

## .

# Divide & Conquer

- No se puede aplicar a todo tipo de problemas.
- Metodología top-down
- Usa recursión en las implementaciones aunque puede aplicarse también de manera iterativa.
- Las soluciones obtenidas se combinan para obtener la solución del problema original.



# Divide & Conquer

#### PASOS:

- 1. Descomponer el problema en k subproblemas del mismo tipo, pero de menor tamaño.
- Resolver independientemente todos los subproblemas, directamente si son elementales o bien de forma recursiva.
- 3. Combinar las soluciones obtenidas para construir la solución del problema original.



# Divide & Conquer

#### **EFICIENCIA:**

- Descomponer una instancia en subinstancias disjuntas y recombinar la solución en forma eficiente.
- 2. Se debe evitar seguir avanzando recursivamente cuando el tamaño de los casos no lo justifique.
- Las subinstancias deben ser del mismo tamaño, tanto como se pueda.

### М

# Divide & Conquer

#### **TEOREMA MAESTRO**

$$T(n)=a T(n/b)+c n^k$$
,  $c\geq 0 \in R$ ,  $a\geq 1 \in R$ ,  $b\geq 2 \in Z$   $y$   $k\geq 0 \in Z$ 

$$T(n) = \Theta(n^{logba})$$
 si  $a > b^k$ 

$$T(n) = \Theta(n^k \log_b n)$$
 si  $a = b^k$ 

$$T(n) = \Theta(n^k)$$
 si  $a < b^k$ 

# Preguntas... ...y a practicar...

