

# Algorithmes diviser-pour-régner - médiane

Soit  $L[1..n]$  une liste d'entiers, et soit  $s$  un entier entre 1 et  $n$ . Le  $s$  ième plus petit entier de  $L$  est défini comme étant l'élément qui serait en position  $s$  si la liste  $L$  était ordonnée de façon non décroissante.

Exemple:  $L = [3, 1, 4, 1, 1, 7]$

3 ième plus petit élément de  $L$  ? 1

$L$  triée =  $[1, 1, 1, 3, 4, 7]$



Étant donné  $L$  et  $s$ , le problème de trouver le  $s$  ième plus petit élément de  $L$  est appelé le problème de sélection.

médiane de  $L$ :  $\lceil n/2 \rceil$  plus petit élément de  $L$

# Algorithmes diviser-pour-régner - médiane

Clairement, on peut utiliser l'algorithme de sélection pour trouver la médiane d'une liste:

On sélectionne tout simplement le  $\lceil n/2 \rceil$  ième plus petit élément de cette liste

On peut aussi faire le contraire i.e. se servir d'une algorithme qui calcule la médiane d'une liste pour faire la sélection du  $s$  ième plus petit élément de cette liste,  $\text{sélection}(L[1..n],s)$ .

- on utilise un algo,  $\text{médiane}(L)$ , qui retourne  $p$  la médiane de la liste  $L$
- on appelle l'algorithme  $\text{pivotbis}(L,p ; k,\ell)$  qui va partitionner  $L$  en 3 sections:
  - $L[1..k]$  qui contient les éléments  $< p$        $s \leq k$        $\text{sélection}(L[1..k],s)$
  - $L[k+1..\ell-1]$  qui contient les éléments  $= p$        $k < s < \ell$        $s=p$
  - $L[\ell..n]$  qui contient les éléments  $> p$        $s \geq \ell$        $\text{sélection}(L[\ell..n],s)$

# Algorithmes diviser-pour-régner - médiane

Peut-on modifier l'algorithme sélection pour ne pas avoir à utiliser la médiane?

**Idée 1:** Utiliser comme quicksort,  $p = L[1]$

- Dans le pire cas, on fait la boucle un nombre quadratique de fois
- On peut montrer que l'algorithme sélection( $L[1..n],s$ ) implémenté avec cette idée est en moyenne en  $\Theta(n)$

Comment éviter le quadratique dans le pire cas?

**Idée 2:** Le nombre de passage dans la boucle demeure logarithmique si on choisit un pivot près de la médiane

# Algorithmes diviser-pour-régner - médiane

Est-on loin de la vraie médiane avec l'algorithme pseudomédian?

**Visualisation:** Supposons qu'on ait  $n=52$  éléments dans la liste  $L$

$$\lfloor n/5 \rfloor = 10$$

