

TP n° 3: Recherche linéaire du k^e plus grand élément

BATOG Guillaume

22 décembre 2006

Un médian d'un ensemble $X = \{e_1, \dots, e_n\}$ de n nombres entiers distincts est un nombre e dans X tels que les nombres d'éléments strictement inférieurs et strictement supérieurs à e diffèrent d'au plus de 1 ($n > 0$). Si n est impair, le médian est unique ; si n est pair, il y a deux médians possibles. Le problème de la sélection consiste à trouver l'élément de rang k dans X , c'est-à-dire l'élément se trouvant en k -ième position quand X est trié en ordre croissant ($1 \leq k \leq n$). Nous supposons l'ensemble X représenté par la liste de ses éléments.

```
type ensemble == int list;;
```

Question 1 Écrire la fonction `card` qui prend un ensemble en argument et retourne son nombre d'éléments. Quel est la complexité de cette fonction par rapport à n ?

```
card : ensemble -> int
```

Soit p un entier quelconque, on sera amené à partitionner l'ensemble X en éléments plus grands ou plus petits que p . p sera appelé le *pivot*.

Question 2 Écrire la fonction `partitionP` (respectivement `partitionG`) qui prend en arguments un entier p et un ensemble X et qui retourne l'ensemble de tous ses éléments strictement plus petits (respectivement strictement plus grands) que p . Quelles sont les complexités de ces fonctions par rapport à n ?

```
partitionP : int -> ensemble -> ensemble
```

Question 3 Écrire la fonction récursive `elementDeRang` qui prend en arguments un nombre entier k et l'ensemble X et qui retourne l'élément de rang k dans X , en choisissant comme pivot le premier élément de la liste représentant X . Quel est le nombre maximal $M(n)$ d'opérations pris par cette fonction ?

```
elementDeRang : int -> ensemble -> int
```

Question 4 En supposant équiprobables tous les rangs possibles pour le pivot choisi dans X , montrer que le temps moyen pris par la fonction précédente est bornée par $T(n)$ où T est une fonction croissante, vérifiant $T(0) = 0$ et la formule de récurrence suivante :

$$T(n) \leq ln + \frac{1}{n} \sum_{i=1}^{n-1} \max \{T(i-1); T(n-i)\}$$

où l est une constante indépendante de n et de k . En déduire une constante c , qu'on déterminera en fonction de l telle que pour tout entier n , on a $T(n) \leq cn$.

On s'intéresse à présent à optimiser le coût dans le cas le pire $M(n)$. Considérons d'abord les sous-ensembles $X_i = \{e_{5i+1}, e_{5i+2}, e_{5i+3}, e_{5i+4}, e_{5i+5}\}$ de 5 éléments consécutifs dans X pour $0 \leq 5i < n$. Soit Y l'ensemble des médians des X_i .

Question 5 Écrire une fonction `medians` qui prend en argument l'ensemble X et qui retourne l'ensemble Y (de cardinal exactement $\lfloor n/5 \rfloor$). Quelle est la complexité de cette fonction ?

`medians : ensemble -> ensemble`

Question 6 Écrire une fonction `elementDeRangBis` qui prend en argument un entier naturel k et l'ensemble X et qui retourne l'élément de rang k dans X , en prenant comme pivot le médian de l'ensemble Y .

`elementDeRangBis : int -> ensemble -> int`

Question 7 Montrer que son temps maximum $M'(n)$ vérifie la formule :

$$M'(n) \leq l'n + M' \left(\left\lfloor \frac{n}{5} \right\rfloor \right) + M' \left(7 \left\lfloor \frac{n}{10} \right\rfloor + 4 \right)$$

et l' une constante indépendante de n et de k . En déduire que, pour n suffisamment grand, on a $M'(n) \leq c'n$ où c' est une constante à déterminer en fonction de l' . Que dire si l'on considère des sous-ensembles de cardinal 3 ? de cardinal impair supérieur à 7 ?

Applications

1. Si $n \geq 140$, alors il y a au moins $\lfloor n/4 \rfloor$ éléments plus grands que le médian des médians x et au moins $\lfloor n/4 \rfloor$ éléments plus petits que x .
2. Le k^{e} quantile d'un ensemble à n éléments est l'ensemble des $k - 1$ rangs qui divisent l'ensemble trié en k sous-ensembles de taille égale (et supérieure à 1). Donner un algorithme en $O(n \log k)$ pour énumérer le k^{e} quantile d'un ensemble.
3. Décrire un algorithme en $O(n)$ qui, étant donné un ensemble S de n nombres distincts et un entier positif $k \leq n$, détermine les k nombres de S qui sont les plus proches du médian de S .
4. Soient deux tableaux de n nombres triés chacun. Donner un algorithme en $O(\log n)$ pour trouver le médian des $2n$ éléments présents dans ces deux tableaux.

L'inspecteur Derrick est consultant pour une compagnie pétrolière, qui projette de construire un grand oléoduc d'Est en Ouest à travers un champ pétrolier de n puits. Chaque puits devra être relié à l'oléoduc principal par un chemin minimal (Nord ou Sud). Connaissant les coordonnées des puits, comment l'inspecteur trouvera (en temps linéaire) l'emplacement de l'oléoduc principal qui optimise la longueur totale des raccordements aux puits ?

