

Partiel du lundi 27 mars 2006

Polycopié du cours autorisé – pas de calculatrice
Le barème est indicatif, il pourra être modifié

Exercice 1 (Notation asymptotique).

3 pt

1. Est-ce que $(n + 3) \log n - 2n = \Omega(n)$?
2. Est-ce que $2^{2n} = O(2^n)$?

(1,5 pt)

(1,5 pt)

Exercice 2 (Complexité en moyenne du tri gnome). *Le but de cet exercice est d'écrire le tri gnome en C et de déterminer le nombre moyen d'échanges effectués au cours d'un tri gnome.*

7 pt

Rappel du cours. « Dans le tri gnome, on compare deux éléments consécutifs : s'ils sont dans l'ordre on se déplace d'un cran vers la fin du tableau (ou on s'arrête si la fin est atteinte); sinon, on les intervertit et on se déplace d'un cran vers le début du tableau (si on est au début du tableau alors on se déplace vers la fin). On commence par le début du tableau. »

1. Écrire une fonction C de prototype `void trigrnome(tableau_t t)` effectuant le tri gnome. (2,5 pt)

Une inversion dans une entrée a_0, \dots, a_{n-1} est la donnée d'un couple d'indices (i, j) tel que $i < j$ et $a_i > a_j$.

Rappel. Un échange d'éléments entre deux indices i et j dans un tableau est une opération qui intervertit l'élément à l'indice i et l'élément à l'indice j , laissant les autres éléments à leur place.

2. Si le tri gnome effectue un échange entre deux éléments, que peut on dire de l'évolution du nombre d'inversions dans ce tableau avant l'échange et après l'échange (démontrer) ? (2,5 pt)

On suppose que le nombre moyen d'inversions dans un tableau de taille n est $\frac{n(n-1)}{4}$.

3. Si un tableau t de taille n contient $f(n)$ inversions, combien le tri gnome effectuera d'échanges sur ce tableau (démontrer) ? En déduire le nombre moyen d'échanges effectués par le tri gnome sur des tableaux de taille n . (2 pt)

Exercice 3 (Tri par base). *Soit la suite d'entiers décimaux 141, 232, 045, 112, 143. On utilise un tri stable pour trier ces entiers selon leur chiffre le moins significatif (chiffre des unités), puis pour trier la liste obtenue selon le chiffre des dizaines et enfin selon le chiffre le plus significatif (chiffre des centaines).*

10 pt

Rappel. Un tri est stable lorsque, à chaque fois que deux éléments ont la même clé, l'ordre entre eux n'est pas changé par le tri. Par exemple, en triant $(2, a), (3, b), (1, c), (2, d)$ par chiffres croissants, un tri stable place $(2, d)$ après $(2, a)$.

1. Écrire les trois listes obtenues. Comment s'appelle cette méthode de tri ? (0,5 pt)

On se donne un tableau t contenant N entiers entre 0 et $10^k - 1$, où k est une constante entière. Sur le principe de la question précédente (où $k = 3$ et $N = 5$), on veut appliquer un tri par base, en base 10 à ces entiers.

On se donne la fonction auxiliaire :

```
int cle(int x, int i){
    int j;
    for (j = 0; j < i; j++)
        x = x / 10; // <- arrondi par partie entière inférieure.
    return x % 10;
}
```

2. Que valent `cle(123, 0)`, `cle(123, 1)`, `cle(123, 2)` (inutile de justifier votre réponse) ?
Plus généralement, que renvoie cette fonction ? (1,5 pt)

On suppose que l'on dispose d'une fonction auxiliaire de tri `void triaux(tableau_t t, int i)` qui réordonne les éléments de `t` de manière à ce que

$$\text{cle}(t[0], i) \leq \text{cle}(t[1], i) \leq \dots \leq \text{cle}(t[N - 1], i).$$

On suppose de plus que ce tri est stable.

3. Écrire l'algorithme de tri par base du tableau `t` (utiliser la fonction `triaux`). On pourra considérer que `k` est un paramètre entier passé à la fonction de tri. (2 pt)
4. Si le temps d'exécution en pire cas de `triaux` est majoré asymptotiquement par une fonction $f(N)$ de paramètre la taille de `t`, quelle majoration asymptotique pouvez donner au temps d'exécution en pire cas de votre algorithme de tri par base ? (1 pt)
5. Démontrer par récurrence que ce tri par base trie bien le tableau `t`. Sur quelle variable faites vous la récurrence ? Où utilisez vous le fait que `triaux` effectue un tri stable ? (3 pt)
6. La fonction `triaux` utilise intensivement la fonction à deux paramètres `cle`. Si on cherche un majorant $f(N)$ au temps d'exécution de `triaux`, peut on considérer qu'un appel à `cle` prend un temps borné par une constante ? (1 pt)
7. Décrire en quelques phrases une méthode pour réaliser la fonction `triaux` de manière à ce qu'elle s'exécute en un temps linéaire en fonction de la taille du tableau (on pourra utiliser une structure de donnée). (1,5 pt)