Partiel du 14 mars 2011

Aucun document autorisé. Le barême est uniquement indicatif.

Vous pouvez écrire les algorithmes en C ou en pseudo-code. Si cela vous pose trop de difficultés n'hésitez pas à répondre en décrivant un algorithme par des phrases ! Vous pouvez faire appel à des fonctions auxilliaires vues en cours (comparaison, tris, sous-tableau, etc.).

Première partie 10 points

Exercice 1 (mars 2011)**.**
Rappeler les définitions utilisées et justifier (démontrer) vos réponses.

1. Est-ce que $n^2 - 2n + 1 = O(n^2)$? 1 pt 9 min
2. Est-ce que $\sum_{i=1}^{n} \log i = \Omega(n)$? 1 pt 9 min
3. Est-il vrai que si $f = O(g)$ et $f = \Omega(h)$ alors $g = \Omega(h)$? 1 pt 9 min

Exercice 2.
Classer les fonctions de complexité $n \log n, 2^n, \log n, n^2, n$ par ordre croissant et pour chacune d'elle donner l'exemple d'un algorithme (du cours ou des TD) qui a asymptotiquement cette complexité en pire cas, en temps. Répondre dans un tableau en donnant le nom de l'algorithme ou le nom du problème qu'il résout. 2 pt 18 min

Exercice 3 (Tri gnome et arbre de décision, meilleur cas)**.** 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 était au début du tableau alors on se déplace d'un cran vers la fin). On commence par le début du tableau. » tot: 5 pt

1. Représenter étape par étape le tri gnome du tableau $[30, 20, 10]$ puis du tableau $[20, 10, 30]$. Vous pouvez ne réprésenter que la position courante (les indices i, $i + 1$ où l'on se trouve), et le contenu du tableau. 1 pt 9 min
2. Dessiner l'arbre de décision du tri gnome sur un tableau de trois éléments $[a, b, c]$. Combien de comparaisons mènent à une situation impossible ? 1 pt 9 min
3. On note $C(N)$ le nombre de comparaisons faites par le tri gnome dans le meilleur des cas sur un tableau de taille N. Quel type de tableau en entrée donne le meilleur cas du tri gnome ? Combien vaut $C(N)$ exactement, en fonction de N ? 1 pt 9 min
4. Écrire une fonction réalisant le tri gnome, en langage C ou en pseudo-code. 2 pt 18 min

Seconde partie : problèmes 11 points

Exercice 4 (Complexité en moyenne du tri gnome)**.**
Le but de cet exercice est de déterminer le nombre moyen d'échanges effectués au cours d'un tri gnome. Rappel : un échange 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. Une *inversion* dans une entrée $a_0, \ldots, a_{n-1}$ est la donnée d'un couple d'indices (i, j) tel que $i < j$ et $a_i > a_j$.

1. Donner la liste de toutes les inversions dans le tableau $t = [8, 10, 9, 6]$ (numéroter les éléments de ce tableau à partir de zéro) (Par exemple $(0, 3)$ est une inversion puisque $t[0] = 8 > t[3] = 6$). Combien y en a t'il ? 0.5 pt 4 min
2. Si au cours du tri d'un tableau, 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 22 min

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 18 min

Exercice 5 (Même problème qu'en mars 2010 !)**.**
Le but de cet exercice est trouver un bon algorithme pour la recherche de valeurs médianes. En économie, le revenu médian est le revenu qui partage exactement en deux la population : la moitié de la population dispose d'un revenu plus élevé que le revenu médian, l'autre moitié d'un revenu moins élevé. Plus généralement, dans un tableau l'élément médian (ou valeur médiane) est l'élément qui serait situé au milieu du tableau si le tableau était trié. Lorsque le nombre d'éléments est pair, il n'y a pas d'élément exactement au milieu. Par convention et pour simplifier, nous prendrons, pour tout N, comme médian l'élément d'indice $\left\lfloor \frac{N}{2} \right\rfloor$ dans le tableau trié (indexé à partir de 0). tot: 6 pt

Par exemple, dans le tableau suivant le revenu médian est de 1200 €.

individus	A	B	C	D	E	F	G	H
revenus mensuels	1400	2000	1300	300	700	5000	1200	800

Pour trouver le médian d'un tableau d'éléments deux à deux comparables on peut trier le tableau puis renvoyer la valeur de son élément d'indice $\left\lfloor \frac{N}{2} \right\rfloor$.

1. Pour cette solution, quelle complexité en moyenne (en temps) peut on obtenir, au mieux ? Quel algorithme de tri choisir ? .5 pt 4 min

On cherche un algorithme plus rapide. Il n'est sans doute pas nécessaire de trier tout le tableau pour trouver le médian. Le professeur Hoare suggère d'utiliser l'algorithme de partition du tri rapide, le fameux *quicksort* (comme en cours, la valeur pivot est choisie dans la première case). Après partition le tableau T est fait de trois partie : la première partie est un tableau T' des éléments de T plus petits que le pivot la deuxième partie ne contient que l'élément pivot et la troisième partie est un tableau T'' des éléments plus grands que le pivot.

2. En fonction de l'indice p du pivot, dans quelle partie chercher le médian ? 1 pt 9 min

En général, on ne trouve pas le médian après le premier partitionnement. L'idée de Hoare est de continuer à partitionner la partie dans laquelle on doit chercher le médian, jusqu'à le trouver.

3. Dans quelle partie chercher le médian à chaque étape ? Répondre en donnant l'algorithme complet (mais en considérant que la fonction de partionnement est fournie). Si nécessaire vous pouvez considérer que la fonction de partionnement renvoie un triplet (T', p, T'') où T' et T'' sont les deux sous-tableaux de T évoqués plus haut et p est l'indice dans T du pivot. 2 pt 18 min

On suppose que le partitionnement d'un tableau de N éléments se fait exactement en N comparaisons ($N-1$ serait également possible). On s'intéresse à la complexité asymptotique de notre algorithme de recherche du médian, exprimée en nombre de comparaisons.

4. Quel est le meilleur cas ? Pour quelle complexité ? Quel est le pire cas ? Pour quelle complexité ? 1 pt 9 min

Pour estimer si la moyenne est plus proche du meilleur cas ou du pire cas, on fait l'hypothèse que chaque fois que l'on fait une partition sur un tableau T, le médian se trouve dans un sous-tableau contenant $\frac{2}{3}$ des éléments de T.

5. Donner un équivalent asymptotique du nombre de comparaisons faites. 1 pt 9 min
6. En supposant que ce résultat représente la complexité moyenne, l'algorithme est-il asymptotiquement optimal en moyenne ? .5 pt 4 min