Devoir sur table

UPMC — Master d'Informatique — STL - M1
UE Algorithmique avancée

29 octobre 2008 – durée 2h

Les seuls documents autorisés sont les polys de cours et de TDs, ainsi que les documents manuscrits.

1 Files binômiales [2 points]

Répondre aux questions suivantes en argumentant les réponses.

1. Quel est le nombre d'arbres binômiaux dans une file binômiale de 743 éléments ?
2. Quel est le nombre de comparaisons nécessaires pour faire la fusion d'une file binômiale de 1221 éléments avec une file binômiale de 829 éléments (expliquer la construction) ?

2 Arbres équilibrés [3 points]

Étant donné un arbre T, il vérifie la propriété $\mathcal{P}$ ssi *il existe une constante c telle que pour tout nœud de T, les hauteurs de ses sous-arbres diffèrent au plus de c.*

1. Montrer que pour tout arbre de taille n et hauteur h vérifiant $\mathcal{P}$, on a $n \geq (1+\alpha)^h - 1$, pour $0 < \alpha \leq 1$.
2. En déduire que tout arbre qui vérifie la propriété $\mathcal{P}$ a une hauteur en $O(\log n)$.

3 Files de priorité [6 points]

Dans cet exercice, on considère les files de priorité représentées par des tas (on rappelle qu'un tas est un arbre binaire croissant dont toutes les feuilles sont situées au plus sur deux niveaux, les feuilles du niveau le plus bas étant calées à gauche).

1. Avec cette structure de données, on peut ajouter un élément ou extraire le minimum en $O(\log n)$ comparaisons dans le cas le pire (où n est la taille du tas). Expliquer brièvement cette assertion.
2. Montrer qu'il est impossible de construire une structure de donnée S représentant une file de priorité, telle que, pour S, on ait à la fois le coût de la construction d'une file de n éléments en O(n) et le coût de la suppression du minimum en $O(1)$.
3. Il s'agit maintenant de montrer que l'on peut extraire le minimum en coût amorti $O(1)$, tout en gardant l'ajout en coût amorti $O(\log n)$. On considère la fonction Φ, qui à un tas T associe

   $$\Phi(T) = 2\sum_{x \in T}(p(x)+1),$$

   où $p(x)$ est la profondeur du nœud x dans l'arbre T.

(a) Montrer que Φ est une fonction de potentiel.
(b) Montrer que le coût amorti de l'ajout d'un élément est en $O(\log n)$.
(c) Montrer que le coût amorti de la suppression du minimum est en $O(1)$.

4 Concaténation et coupure d'arbres 2-3-4 [6 points]

Cet exercice travaille sur les arbres 2-3-4 à éléments distincts, comme dans le Devoir Maison ; vous devez donner les spécifications des primitives utilisées dans les algorithmes.

1. Écrire un algorithme qui, étant donné deux arbres 2-3-4 A et B, tels que tous les éléments de A sont inférieurs aux éléments de B, et un élément x strictement compris entre les éléments de A et ceux de B, construit un arbre 2-3-4 formé de $A \cup \{x\} \cup B$.
   Quelle est la complexité de cet algorithme ?
2. Étant donné un arbre 2-3-4 C et un élément x dans C, il s'agit d'effectuer la coupure de C selon x, en renvoyant deux arbres 2-3-4 A et B, tels que A contient tous les éléments de C inférieurs à x, et B contient tous les éléments de C strictement supérieurs à x.
   Expliquer les différents cas de figures possibles, écrire l'algorithme de coupure et donner l'ordre de grandeur de sa complexité.

5 Hachage universel [3 points]

Soit $\mathcal{H}$ une classe de fonctions de hachage dans laquelle chaque fonction $h \in \mathcal{H}$ envoie l'univers de clefs U sur les entiers $0, 1, \ldots, m-1$. On rappelle que $\mathcal{H}$ est dite *k-universelle* ssi, pour toute séquence de k clefs distinctes $(x_1, \ldots, x_k)$, pour toute fonction h choisie au hasard dans $\mathcal{H}$ et pour toute suite de valeurs $m_1, \ldots, m_k$ dans $[0, 1, \ldots, m-1]$,

$$\Pr(h(x_1) = m_1, \ldots, h(x_k) = m_k) = \frac{1}{m^k}.$$

On considère l'univers U des n-uplets de valeurs de $\mathbb{Z}_p = \{0, 1, \ldots, p-1\}$, où p est premier.
Soit $x = \langle x_0, \ldots, x_{n-1} \rangle \in U$, pour tout n-uplet $a = \langle a_0, \ldots, a_{n-1} \rangle \in U$ et $b \in \mathbb{Z}_p$, on définit la fonction de hachage $h_{a,b}$ par :

$$h_{a,b}(x) = \left(\Sigma_{j=0}^{n-1} a_j x_j + b\right) \; mod \; p.$$

Soit $\mathcal{H} = \{h_{a,b} \; ; a \in U, \; b \in \mathbb{Z}_p\}$.

1. Montrer que $|\mathcal{H}| = p^{n+1}$.
2. Montrer que $\mathcal{H}$ est 2-universelle.
3. La famille $\mathcal{H}$ est-elle 3-universelle ?