Université de Marne-la-Vallée

Licence d'informatique et d'I.U.P.
Examen d'algorithmique

18 décembre 1998, durée : 3h

Seule la consultation des supports de cours, travaux dirigés et travaux pratiques est autorisée pendant le contrôle. Chaque question compte pour deux points.

Huffman

1. Appliquer l'algorithme de Huffman en remplissant la table ci-dessous. Dessiner l'arbre obtenu en notant sur chaque nœud l'indice qui lui correspond.

	0	1	2	3	4	5	6	7	8	9	10	11	12
poids	8	5	3	2	1	1	1						
gauche	-1	-1	-1	-1	-1	-1	-1						
droit	-1	-1	-1	-1	-1	-1	-1						

2. On considère une suite $\mathcal{F}$ de cinq feuilles de poids respectifs 1,1,1,2,3. Donner les différents arbres de Huffman construits sur $\mathcal{F}$ (sans détailler leur construction). Donner le poids et la longueur externe pondérée ($\sum_{f\in\mathcal{F}} p(f)niv(f)$, où $p(f)$ est le poids de la feuille f et $niv(f)$ son niveau) des arbres.

On considère la suite $G = (G_k \mid k \in \mathbf{N})$ définie par $G_0 = G_1 = G_2 = 1$ et par $G_k = G_{k-1}+G_{k-2}$ pour $k \geq 3$.

3. Quels sont les poids des arbres de Huffman construits sur les suites $(G_0,G_1,\ldots,G_q)$ pour $0 \leq q \leq 7$?
Soit $q \in \mathbf{N}$. Quel est le poids d'un arbre de Huffman contruit sur la suite $(G_0,G_1,\ldots,G_q)$?

4. Soit $q \in \mathbf{N}$. Écrire un algorithme qui construit un arbre de Huffman de hauteur maximum sur la suite $(G_0,G_1,\ldots,G_q)$. [On effectuera une preuve par invariant pour montrer que l'algorithme construit bien un arbre de Huffman et que celui-ci est de hauteur q.]

Rotations

Dans cet exercice on considère des arbres binaires de recherche. Un arbre non vide A est noté $(a,\mathcal{G}(A),\mathcal{D}(A))$ où a est la racine de l'arbre, $\mathcal{G}(A)$ son sous-arbre gauche et $\mathcal{D}(A)$ son sous-arbre droit. Chaque nœud b des arbres considérés possède une étiquette notée $Elt(b)$ qui est un entier. On note $h(A)$ la hauteur de A (on rappelle que la hauteur d'un arbre vide est -1, par convention). On note $\Delta(A) = h(\mathcal{D}(A)) - h(\mathcal{G}(A))$, l'équilibre de A.

Pour un arbre binaire $X = (x,(y,U,V),W)$ (x,y sont des nœuds et U,V,W des arbres binaires) on définit $\textsc{Rotd}(X) = (y,U,(x,V,W))$ le résultat d'une rotation droite appliquée à X. On remarque que $\textsc{Rotd}(A)$ n'est défini que pour un arbre binaire A qui possède un sous-arbre gauche non vide.

5. Écrire en langage C une déclaration d'arbres binaires de recherche et une implémentation de la fonction $\textsc{Rotd}$.

6. Montrer que si X est un arbre binaire de recherche sur l'ensemble des étiquettes de ses nœuds, il en est de même de $\textsc{RotD}(X)$.

Soit $X = (x,(y,U,V),W)$ comme ci-dessus. On note $u = h(U)$, $v = h(V)$, $w = h(W)$, ainsi que $p = \Delta(X) = w - \max(u,v) - 1$, et $q = \Delta(\mathcal{G}(X)) = \Delta((y,U,V)) = v - u$.

7. Exprimer $q' = \Delta(\mathcal{D}(\textsc{RotD}(X)))$ en fonction de p et q (sans u, v, ni w). On pourra distinguer le cas $q \geq 0$ du cas $q < 0$, puis regrouper les deux expressions en une seule.

8. Exprimer $p' = \Delta(\textsc{RotD}(X))$ en fonction de p et q (sans u, v, ni w). On pourra distinguer trois cas suivant les signes de q' et q. Montrer que p' s'écrit sous la forme $\max\{e_1,e_2,e_3\}$ où e_1,e_2,e_3 sont des expressions en p et q (sans u, v, ni w).

Un arbre binaire de recherche A est dit AVL si :

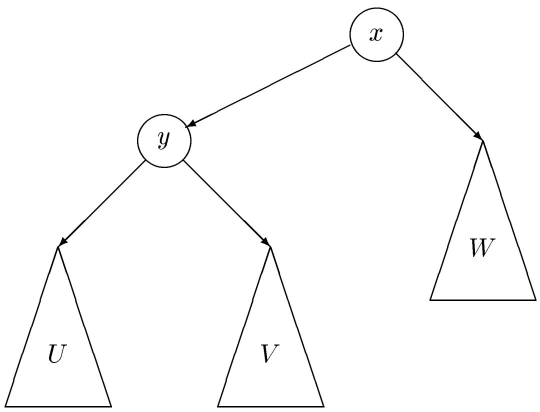
- A est l'arbre vide, ou
- A est non vide, $\mathcal{G}(A)$ et $\mathcal{D}(A)$ sont AVL, et $-1 \leq \Delta(A) \leq 1$.

9. Soit A un arbre binaire de recherche dont le sous-arbre gauche est non vide. On suppose $\Delta(A) = -2$ et $\Delta(\mathcal{G}(A)) = -1$.
Montrer que $h(\textsc{RotD}(A)) = h(A) - 1$.
Montrer que $\textsc{RotD}(A)$ est AVL si $\mathcal{G}(A)$ et $\mathcal{D}(A)$ sont AVL.

10. Soit $\textsc{Ajout}(X,z)$ l'arbre binaire de recherche obtenu par ajout de l'élément z à l'arbre binaire de recherche X au moyen de la procédure usuelle. On s'intéresse à l'une des situations qui nécessite un rééquilibrage de l'arbre $\textsc{Ajout}(X,z)$.
On suppose que X est AVL et que $\textsc{Ajout}(X,z)$ possède un sous-arbre A pour lequel $\mathcal{G}(A)$ et $\mathcal{D}(A)$ sont AVL, $\Delta(A) = -2$, et $\Delta(\mathcal{G}(A)) = -1$.
Montrer que l'arbre obtenu à partir de $\textsc{Ajout}(X,z)$ en remplaçant A par $\textsc{RotD}(A)$ est AVL.

11. Quel peut être le temps d'exécution de l'ajout d'un élément dans un arbre AVL ? Justifier la réponse.

L'arbre $X = (x,(y,U,V),W)$



L'arbre $\textsc{RotD}(X) = (y,U,(x,V,W))$