Université de Marne-la-Vallée

Licence d'informatique et d'I.U.P.
Examen de **Structure de données**

Janvier 2002, durée : 2h

Seule la consultation des supports de cours, travaux dirigés et travaux pratiques est autorisée pendant le contrôle.

Compression

On considère le codage de Huffman appliqué au mot $t =$ "`abraracourci`".

1) Dessiner l'arbre de Huffman associé au codage des lettres de t. Donner la représentation par curseur de votre arbre dans une table de la forme :

lettre	**a**	**r**	**c**	**b**	**i**	**o**	**u**						
occurr.	3	3	2	1	1	1	1						
gauche	-1	-1	-1	-1	-1	-1	-1						
droit	-1	-1	-1	-1	-1	-1	-1						

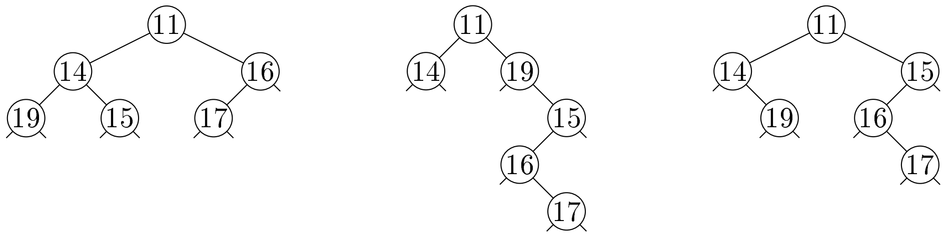
2) On note $C(a)$ le code ASCII d'un caractère a. Donner un procédé pour mémoriser un arbre de Huffman dans un fichier. Indiquer le contenu de ce fichier pour l'arbre de la question précédente.

Tournois

On considère des arbres binaires dont les nœuds sont étiquetés par un ensemble fini X d'entiers naturels ($X \subseteq \mathbb{N}$). L'étiquette d'un nœud s est notée $val(s)$ et l'arbre binaire vide est noté NIL. Si T est un arbre binaire non vide ses sous-arbres gauche et droit sont notés $\mathcal{G}(T)$ et $\mathcal{D}(T)$ respectivement.

Un arbre binaire T sur X est un *tournoi sur* X si
— soit $T = \text{NIL}$ et X est vide,
— soit $T = (r, \mathcal{G}(T), \mathcal{D}(T))$ avec $X = \{val(s) \mid s \text{ nœud de } T\}$, $val(r) \leq val(s)$ pour tout nœud s de T, et les sous-arbres gauche et droit $\mathcal{G}(T)$ et $\mathcal{D}(T)$ sont des tournois.

3) Quels sont ceux des arbres des figures (a), (b) et (c) qui sont des tournois ?



(a) (b) (c)

Donner la liste des nœuds en ordre préfixe pour chacun de ces arbres. Montrer que la liste des nœuds en ordre préfixe ne caractérise pas nécessairement un tournoi.

On définit l'*insertion* d'un élément x dans un tournoi T par :

$$Ins(T, x) = \begin{cases} (s, \text{NIL}, \text{NIL}) \text{ avec } val(s) = x, & \text{si } T = \text{NIL}, \\ (s, T, \text{NIL}) \text{ avec } val(s) = x, & \text{si } T = (r, \mathcal{G}(T), \mathcal{D}(T)) \text{ et } x \leq val(r), \\ (r, \mathcal{G}(T), Ins(\mathcal{D}(T), x)), & \text{si } T = (r, \mathcal{G}(T), \mathcal{D}(T)) \text{ et } x > val(r). \end{cases}$$

4) Quel est le tournoi construit par insertions successives des éléments

$$11, \ 5, \ 3, \ 6, \ 8, \ 15,$$

dans un tournoi initialement vide ? Faire figurer le résultat obtenu après chaque insertion.
Montrer que si T est un tournoi sur X et $x \in \mathbb{N}$, $Ins(T, x)$ est un tournoi sur $X \cup \{x\}$.

La *distance* d'un arbre binaire T est définie par :

$$dist(T) = \begin{cases} 0, & \text{si } T = \text{NIL}, \\ 1 + \min\{dist(\mathcal{G}(T)), dist(\mathcal{D}(T))\}, & \text{si } T = (r, \mathcal{G}(T), \mathcal{D}(T)). \end{cases}$$

On dit d'un tournoi T qu'il est *gauchiste* si
— soit $T = \text{NIL}$,
— soit $T = (r, \mathcal{G}(T), \mathcal{D}(T))$ avec $dist(\mathcal{G}(T)) \geq dist(\mathcal{D}(T))$, et les sous-arbres gauche et droit $\mathcal{G}(T)$ et $\mathcal{D}(T)$ sont des tournois gauchistes.

On adopte pour les tournois gauchistes les types de données suivants :

```
typedef struct NOEUD {
    int val, dist;
    struct NOEUD * g, * d;
 } noeud;
typedef noeud * tournoi;
```

5) On considère un tournoi T pour lequel les membres `dist` des nœuds sont nuls. Écrire en langage C une fonction d'en-tête

`void Init_dist(tournoi T)`

qui, agissant sur T déjà construit, initialise les membres `dist` de chaque nœud s de T (c'est-à-dire donne au membre `dist` de chaque nœud s la valeur de la distance du sous-arbre de racine s). Quel est le temps d'exécution de votre fonction ?

6) Le tournoi construit à la question **4** est-il gauchiste ? Faire figurer sous chacun de ses nœuds la distance du sous-arbre dont il est la racine. Montrer que la condition $dist(\mathcal{G}(T)) \geq dist(\mathcal{D}(T))$ de la définition ci-dessus est équivalente à $dist(T) = 1 + dist(\mathcal{D}(T))$.

7) Soit T un tournoi gauchiste. Montrer que si $dist(T) = k$ ($k \in \mathbb{N} \setminus \{0\}$) l'arbre T possède au moins 2^{k-1} feuilles et $2^k - 1$ nœuds. Montrer que si T possède n nœuds on a :

$$dist(T) \leq \log_2(n+1).$$