Université de Marne-la-Vallée

Licences d'informatique et d'I.U.P.
Examen de **Structures de données**

Janvier 2003, durée : 2h

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

B-arbres

1) Donner les propriétés qui caractérisent les B-arbres.

2) Après avoir donné une valeur aux paramètres de la définition, décrire en langage C des structures de données pour implémenter des B-arbres.

Parcours

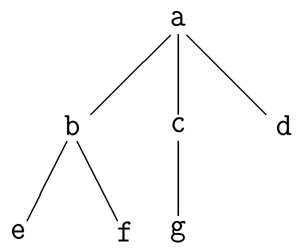
Le but de cet exercice est l'étude d'un algorithme de parcours d'arbre planaire qui ne nécessite pour fonctionner qu'un espace mémoire de taille constante en plus de l'espace occupé par l'arbre. (Les algorithmes usuels utilisent une pile qui occupe un espace mémoire dont la taille dépend de la hauteur de l'arbre.)

Une étape de l'algorithme du parcours (transformation T) transforme l'arbre non vide $\mathcal{A}$ en l'arbre $T(\mathcal{A})$. Si $\mathcal{A}$ ne possède qu'un nœud, $T(\mathcal{A}) = \mathcal{A}$. Sinon, $\mathcal{A}$ possède au moins un sous-arbre (non vide) et est de la forme $\mathcal{A} = (r, \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_k)$ avec $k > 0$. Le sous-arbre $\mathcal{A}_1$ non vide est de la forme $\mathcal{A}_1 = (s, \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_\ell)$ avec $\ell \geq 0$. Au moyen de ces notations l'arbre $T(\mathcal{A})$ est alors

$$T(\mathcal{A}) = (s, \mathcal{B}_1, \mathcal{B}_2, \ldots, \mathcal{B}_\ell, (r, \mathcal{A}_2, \ldots, \mathcal{A}_k)).$$

3) Dessiner l'arbre obtenu par application de la transformation T à l'arbre de la figure ci-dessous.

4) Itérer autant de fois que nécessaire la transformation T jusqu'à retrouver l'arbre de la figure ci-dessous.



Soit $(\mathcal{A}_i \mid i \geq 0)$ la suite d'arbres définie par : $\mathcal{A}_0 = \mathcal{A}$, et $\mathcal{A}_i = T(\mathcal{A}_{i-1})$ pour $i > 0$. Soit $m(\mathcal{A})$ l'entier i non nul minimal tel que $\mathcal{A}_i = \mathcal{A}$.

5) Montrer que chaque nœud de l'arbre $\mathcal{A}$ est racine d'un des arbres $\mathcal{A}_0$, $\mathcal{A}_1$, $\ldots$, $\mathcal{A}_{m(\mathcal{A})-1}$.

6) Quelle est la valeur de $m(\mathcal{A})$ si $\mathcal{A}$ possède n nœuds ?

7) Décrire en langage C des structures de données adaptées aux arbres planaires sur lesquels on veut appliquer la transformation T.

8) Écrire en langage C une fonction qui réalise la transformation T en utilisant les structures de données décrites dans la question précédente.

9) Indiquer comment détecter la fin du parcours complet de l'arbre $\mathcal{A}$, c'est-à-dire la $m(\mathcal{A})$-ième étape du parcours, lorsqu'on ne connaît ni $m(\mathcal{A})$ ni la taille de l'arbre $\mathcal{A}$. Décrire l'algorithme de parcours complet d'un arbre planaire.

10) Quel est le temps d'exécution de votre implémentation de la transformation T et celui du parcours complet ?