

EXAMEN DU MODULE IF102

Lundi 19 janvier 2004

Durée 2h, documents autorisés, livres interdits

1. Considérant un ensemble à n éléments munis d'une clef, on définit son i ème rang, où $i \in \{1, \dots, n\}$, comme étant l'élément ayant la i ème plus petite clef.

Si l'on suppose que l'ensemble n'est pas trié, quelle est la complexité dans le pire des cas de la recherche d'un rang quelconque? Justifiez votre réponse.

2. Rappeler la définition des arbres rouge et noir et quelle est leur utilité. Proposer un type abstrait et une implémentation qui servira à l'écriture des fonctions demandées dans les questions suivantes.
3. On considère des arbres rouge et noir avec des informations supplémentaires conservées en chaque noeud. En plus des champs clef, gauche, droit, parent et couleur, tout noeud x contient aussi un champ taille. Ce champ contient le nombre de noeuds du sous-arbre enraciné en x . Ces arbres seront appelés arbres de rang dans la suite.

Avant de considérer les opérations de suppression et d'insertion dans les arbres de rang, nous allons nous intéresser à la recherche d'un élément d'un rang donné. Écrire une procédure RECUPERER-RANG(x, i) retournant une référence sur le noeud contenant la i ème plus petite clef du sous-arbre enraciné en x . Écrire une version récursive et une version non récursive de cette opération. Quelle est la complexité de cette procédure pour un ensemble de n éléments?

4. Écrire une procédure DETERMINER-RANG(T, x) retournant la position de x dans l'ordre linéaire déterminé par un parcours préfixe de l'arbre de rang T . Quelle est la complexité de cette procédure pour un ensemble de n éléments?
5. Écrire les opérations de rotation agissant sur les noeuds d'un arbre de rang et mettant à jour le champ taille. Calculer la complexité de ces opérations.
6. Expliquer le principe de l'insertion dans un arbre de rang à partir de celui de l'insertion dans l'arbre rouge et noir simple. Écrire l'opération d'insertion dans un arbre de rang de façon à conserver la même complexité que pour un arbre rouge et noir simple. Montrer que la complexité est bien la même.
7. Expliquer le principe de la suppression dans un arbre de rang à partir de celui de la suppression dans l'arbre rouge et noir simple. Écrire l'opération de suppression dans un arbre de rang de façon à conserver la même complexité que pour un arbre rouge et noir simple. Montrer que la complexité est bien la même.