Exercice 1. Rappels sur les ABR

Etudier successeur, predecesseur, recherche, insertion et suppression.

Exercice 2. Final printemps 2003

Soit A un arbre binaire de recherche (ABR) et a une valeur. On veut déterminer la plus petite valeur b contenue dans A qui est inférieure ou égale à a. Si a est supérieur au maximum de A, on affiche un message d'erreur.

Par exemple, A contient les valeurs 1, 3, 5, 7, 9. Si a = 4 alors b = 5. Si a = 9 alors a = b = 9. Si a = 10 alors on affiche un message d'erreur.

Un nœud de l'arbre A correspond à une structure de type structnoeud comportant trois champs : une valeur de type int, et deux pointeurs sur le nœud gauche et le nœud droit. Le type arbre utilisé dans toute la suite sera donc un pointeur sur la structure structnoeud.

- **A.** Définir la structure structnoeud et le type arbre.
- **B.** Ecrire une fonction max en C qui retourne la valeur maximale contenue dans un ABR non vide, ainsi qu'une fonction min qui retourne la valeur minimale :

- C. Ecrire un algorithme récursif succ(A, a) qui prend en entrée un ABR A et une valeur a et qui retourne la plus petite valeur b dans l'ABR A qui est supérieure ou égale à a (ou un message d'erreur, si une telle valeur n'existe pas).
- **D.** Ecrire une fonction succ(arbreA, inta) en C qui correspond à l'algorithme que vous proposez dans la question précédente.

Exercice 3.

On s'intéresse à la gestion informatique des résérvations sur l'année d'une salle de conférences. La salle ne peut être réservée que par journées. Ainsi, une réservation peut-être assimilée à un intervalle de jours I = [u(I), v(I)], où u(I) est le jour d'arrivée des participants et v(I) le dernier jour de la conférence. Pour un intervalle I, on a trois opérations de base : ajouter(I), supprimer(I) et intersecte(I) (qui renvoie VRAI lorsqu'il existe une réservation intersectant I).

- **A.** Représenter graphiquement un arbre binaire de recherche (champs, clés?) associé aux réservations successives [37,39], [3,7], [100,120], [50,68], [10,30], [1,2], [8,9]. Supprimer graphiquement la réservation [3,7].
- **B.** Décrire et prouver l'algorithme de la fonction intersecte. Analyser sa complexité.
- **C.** En supposant que les réservations acceptées sont gérées à l'aide d'une liste chaînée, indiquer la complexité des trois opérations de base lorsque la liste est quelconque puis lorsqu'elle est est triée selon les u(I) croissants. Comparer.