

TD numéro 3

Arbres Binaires de Recherche

Programmation avancée, ENSIIE

Semestre 2, 2010–11

Exercice 1 : Arbres binaires de recherche

1. Montrer que si on affiche un arbre binaire de recherche suivant le parcours infixe, les éléments sont donnés par ordre croissant des clefs.
2. Montrer que `insérer` préserve l'invariant des arbres binaires de recherche.
3. Montrer que si `k` n'est pas dans `d`, alors
 - `rechercher(insérer(d,k,v),k) = v` et
 - `rechercher(insérer(d,k,v),k') = rechercher(d,k')` pour $k \neq k'$.
4. Écrire la fonction `supprimer` en vous servant éventuellement des fonctions `le_plus_a_droite` et `supprimer_la_racine` (à écrire elles aussi le cas échéant).

Exercice 2 : Arbres AVL

1. Montrer que les rotations préservent l'invariant des arbres binaires de recherche.
2. En supposant que ses sous-arbres sont bien équilibrés et que leur différence de hauteur est d'au plus 2, montrer qu'appliquer la fonction `equilibrer` à un arbre retourne un arbre équilibré.
3. Que faut il modifier dans la fonction `supprimer` de la question 4 de l'exercice 1 ?