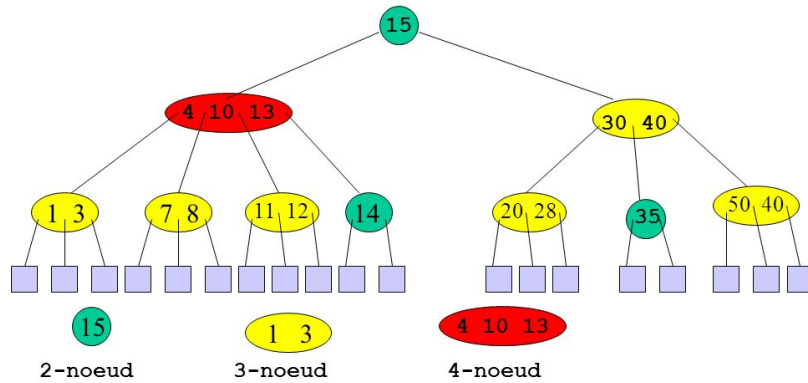


Arbres 2.3.4



Définition : Un **arbre 2.3.4** est un arbre de recherche dont les nœuds sont de trois types, 2-nœud, 3-nœud, 4-nœud, et dont toutes les feuilles sont situées au même niveau

78

Compte Rendu TP Algorithmique Avancée n°2

ARBRE 234, COLORIÉS

Mottino Loris | Da Costa Tom | Verrier Martin

Introduction :

Lien GitHub : https://github.com/iogepryga/ALG_TP2.git

Lors de ce TP nous allons manipuler une nouvelle structure de donnée : les arbres 234, qui est un arbre un peu particulier avec des 0noeud 2noeud 3noeud 4noeud.

Explication de notre Code :

Nous vous invitons à aller voir notre code qui est commenté.

Nous allons tout de même expliquer quelques fonctions :

int NombreCles (Arbre234 a) :

La fonction est codée en récursivité postfixe, avec un parcours en profondeur de l'arbre.

On fixe la condition d'arrêt tel que : `a == NULL` ou `a` est une feuille, c'est-à-dire, `a` est un nœud avec 0 clé.

Il est important de noter que cette fonction parcourt en profondeur l'arbre jusque dans les feuilles. De plus il faut distinguer 3 cas particuliers :

1. Si on est sur un 2noeud : On fait un appel récursif sur ses 2 fils aux indices 1 et 2 du array "fils" puis on ajout 1 (un 2noeud à 1 clé).
2. Si on est sur un 3 ou 4noeud : On fait un appel récursif sur leur fils qui sont cette fois-ci rangé dans l'ordre dans l'array "fils" donc on boucle simplement sur les fils.
3. Si on est sur un 0noeud, une feuille sans clé : s'il n'y a pas de clé, alors on return 0.

Arbre234 RechercherCle (Arbre234 a, int cle) :

La fonction est codée de manière itérative.

Le principe est le suivant : on cherche la position de la clé parmi les clés du nœud exploré actuellement, grâce à ça on sait alors dans quel fils aller ensuite.

On s'arrête dès qu'on a atteint le fond de l'arbre ($a \rightarrow t == 0$) ou alors lorsqu'on a trouvé la clé.

On a alors plusieurs cas :

- Dans le cas d'un 2noeud : si la clé recherchée est plus grande que la clé du nœud exploré : on explore le fils "à droite" sinon celui "de gauche"
- Pour un 3 et 4 nœud, on boucle sur les clés, puis dès qu'on trouve une clé du nœud supérieur à celle recherchée, on explore le fils "juste à gauche" (C'est à dire celui qui contient des clés comprises entre la clé du nœud d'avant (itération précédente) et la clé du nœud étudié actuellement).
- Si aucune des clés du nœud n'est supérieur (on a atteint la dernière clé ($i == \text{derniere_cles}(a)$)), alors on explore le dernier fils ("le plus à droite").

Conclusion :

Lors de ce TP nous avons appris à manipuler une structure de donné arbre 234. Nous avons manipulé cette structure à travers différentes fonctions de parcours, de recherche, et de modification de l'arbre (destruction clé). Nous avons bien manipulé des fonctions récursives qu'itératives.