

TD-TP2 Algorithmique Avancée

Polytech Grenoble, département INFO

Jean-François Méhaut

11 février 2020

Le TP2 d'Algo est à rendre avant le Vendredi 13/3/2020. Vous déposerez sur le moodle une archive contenant le code source et un rapport de 4 pages décrivant le travail que vous avez effectué. Ne mettez pas dans l'archive les fichiers objets (.o) et les exécutables. Je recompilerais avec les fichiers **Makefile**.

Vous rendrez une archive (.tar.gz) avec les fichiers abr.c, avl.c, des jeux de données (fichiers avec des clés) ainsi qu'un texte décrivant le principe des algos que vous avez implémentés.

1 Préparation du TP

Vous allez récupérer l'archive **A234.tar.gz** sur le moodle.

Vous allez ensuite exécuter les commandes suivantes.

```
$ tar xvfz A234.tar.gz
A234/
A234/lire_affichage.c
A234/data/
A234/data/arbre15
A234/data/arbre8
A234/data/arbre1~
A234/data/arbre2
A234/data/arbre14
A234/data/arbre6
A234/data/arbre7
A234/data/arbre1
A234/data/arbre13
A234/data/arbre4
A234/data/arbre11
A234/data/arbre10
A234/data/arbrecours
A234/data/arbre16
A234/data/arbre5
A234/data/arbre3
A234/data/arbre12
A234/data/arbre9
A234/Makefile
A234/a234.c
A234/a234.h
A234/ajout_cle.c
$ cd A234
```

```

$ make
gcc -Wall -g -c a234.c
gcc -Wall -g -c lire_affichage.c
gcc -Wall -g -c ajout_cle.c
gcc -o a234 a234.o lire_affichage.o ajout_cle.o
$ a234 data/arbrecours
===== Afficher arbre =====

                35 40 50 (2)
            30 (1)
                20 28 (2)
10 15 (0)
                13 14 (2)
            12 (1)
                11 (2)
                4 7 8 (2)
            3 (1)
                1 (2)

$

```

2 Arbres A234

Le fichier `a234.h` contient la déclaration du type `Arbre234`.

Le fichier `lire_affichage.c` contient les procédures de lecture et d'affichage des arbres 234.

Le fichier `ajout.c` contient la fonction d'ajout d'une clé dans un arbre 234.

Un fichier `makefile` est fourni pour compiler l'ensemble des fichiers. Vous le complétez si vous ajoutez des nouveaux fichiers.

Pour ce TP, vous devez compléter ce code avec les procédures/fonctions suivantes.

Je vous suggère de traiter ces procédures/fonctions dans l'ordre qui est indiqué (des plus faciles aux plus difficiles).

1. Complétez la fonction `NombreClés` qui calcule le nombre de clés d'un arbre 234.

```

int NombreCles (Arbre234 a)

    /*
       Retourne le nombre de clés présentes dans l'arbre a
    */

```

2. Complétez la fonction **CleMax** qui calcule la plus grande clé d'un arbre 234.

```
int CleMax (Arbre234 a)

/*
    plus grande cle de l'arbre a
*/
```

3. Complétez la fonction **CleMin** qui calcule la plus petite clé d'un arbre 234.

```
int CleMin (Arbre234 a)

/*
    Retourne la plus petite cle de l'arbre a
*/
```

4. Complétez la fonction **RechercherCle** qui retourne le nœud qui contient la clé passée en paramètre.

```
Arbre234 RechercherCle (Arbre234 a, int cle)

/*
    rechercher si la cle a est presente dans
    l'arbre a. Si oui, retourne le noeud ou se trouve la cle.
*/
```

5. Complétez la fonction **AnalyseStructureArbre** qui retourne en paramètre le nombre de feuilles, le nombre de 2-nœuds, le nombre de 3-nœuds et le nombre de 4-nœuds.

```
void AnalyseStructureArbre (Arbre234 a, int *feuilles, int *noeud2,
                           int *noeud3, int *noeud4)

/*
    calculer le nombre de feuilles, de 2-noeuds, 3-noeuds, et 4-noeuds
*/
```

6. Complétez la fonction **noeud_max** qui retourne le nœud avec les clés dont la somme est maximale.

```
Arbre234 noeud_max (Arbre234 a)

/*
    Retourne le noeud avec la somme maximale des cles internes
*/
```

7. Complétez la fonction **Afficher_cles_largeur** qui affiche les clés en parcourant l'arbre234 en largeur.

```
void Afficher_Cles_Largeur (Arbre234 a)

/*
    Afficher le clés de l'arbre a avec
    un parcours en largeur
*/
```

8. Complétez la fonction récursive **Afficher_cles_triees_recursive** qui affiche les clés en ordre croissant.

```
void Affichage_Cles_Triees_Recursive (Arbre234 a)
/*
    Afficher les clés en ordre croissant de l'arbre a
    Cette fonction sera recursive
*/
```

9. Complétez la fonction nonrécursive **Afficher_cles_triees_nonrecursive** qui affiche les clés en ordre croissant.

```
void Affichage_Cles_Triees_NonRecursive (Arbre234 a)
/*
    Afficher les clés en ordre croissant de l'arbre a
    Cette fonction ne sera pas recursive
    Utiliser une pile
*/
```

10. Complétez la fonction **detruire_cle** qui détruit la clé passée en paramètre de l'arbre 234 a.

```
void Detruire_Cle (Arbre234 *a, int cle)

/*
    retirer la cle de l'arbre a
*/
```