

Algorithmique : TD 6

Licence M/I

31 octobre 2005

Exercice 1: On considère l'algorithme suivant :

```
static Fifo maProcedure(Fifo file,int size){
    Lifo pile=new ArrayLifo(size);//avec size>= taille de file
    boolean r=true;
    Integer el,m,min=new Integer(Integer.MIN_VALUE);
    m=min; r=false;
    while(true){
        el=(Integer) file.get();
        if(el==min){
            if (!r) break; else r=false; //Controle
        }
        if(el.compareTo(m)>0 || el==min){
            file.put(m);
            while(!pile.empty())
                file.put(pile.get());
        }
        else{
            pile.put(m);
            r=true;
        }
        m=el;
    }
    file.put(m);
    return file;
}
```

Exécuter cette fonction sur une file contenant six entiers distincts. Que fait-elle? Quel est le principe de l'algorithme. Donner une famille de files telles que le nombre de fois où la ligne **Controle** est exécutée est équivalent au nombre d'éléments de la file.

Exercice 2: On représente un arbre comme un ensemble de nœuds qui pointent vers leur père. Les feuilles de l'arbre sont stockées dans l'ordre dans une file. Décrire un algorithme qui retourne le mot correspondant au parcours préfixe de l'arbre. Comment obtenir le mot postfixe ?

Exercice 3: Existe-t-il un arbre à 8 sommets dont le parcours préfixe est 12345678 et le parcours postfixe est

53247681?

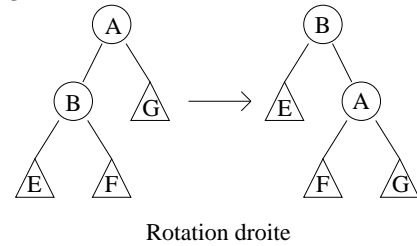
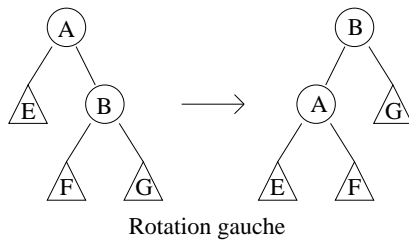
43527861?

Donner un algorithme pour reconstruire un arbre binaire à partir de ses mots préfixe et postfixe.

Exercice 4: Un AVL est un arbre binaire (de recherche) tel que, en chaque nœud, la différence de hauteur des sous-arbres gauche et droit est au plus égale à 1. Dessiner un AVL de hauteur 5 avec un nombre de nœuds minimal. Montrer que le nombre minimal H_n de nœuds d'un AVL de

hauteur n vérifie la récurrence $H_{n+1} = H_n + H_{n-1} + 1$. En déduire que H_n est borné inférieurement par $\left(\frac{1+\sqrt{5}}{2}\right)^n$ pour n assez grand.

Pour implanter des AVL, on stocke dans chaque nœud la différence de hauteur entre le sous-arbre gauche et le sous-arbre droit. Un ABR est un AVL si et seulement si cette différence est comprise entre -1 et 1 pour chaque nœud. Pour modifier l'équilibre d'un nœud, on peut procéder à des *rotations*. Les deux rotations de base sont la rotation à gauche et la rotation à droite :



Ces opérations conservent les propriétés d'arbre binaire de recherche.

Écrire la fonction d'insertion d'une valeur dans un AVL.

Écrire la fonction de suppression d'une valeur dans un AVL.