Université Paris XII
IUT de Fontainebleau
Département informatique
FI-2
2001-2002

PARTIEL 1

Seuls sont autorisés, à titre de documents, les listings comprenant explicitement le nom de l'étudiant (et celui-ci seulement) et les notes manuscrites.

Exercice 1.- (**Arbre binaire de tri**)

Un **arbre binaire** *est une structure de donnée dynamique définie récursivement : on part d'un nœud appelé* **racine**, *chaque nœud comprend une donnée, un* `fils gauche` *et un* **fils droit**, *chacun étant vide ou étant lui-même un nœud. Un nœud sans fils est appelé une feuille.*

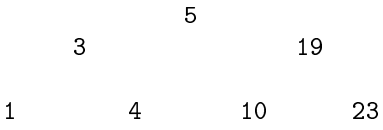
- 1^o) Définir les classes `noeud` des éléments d'un arbre binaire d'entiers naturels et `arbre` en Java.

La classe `noeud` comprendra trois données (un entier, qui est sa valeur, et deux nœuds : le fils gauche et le fils droit) et un constructeur ayant un argument entier permettant de créer un nœud feuille dont la valeur est cet entier.

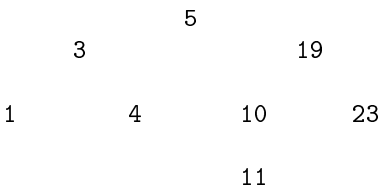
La classe `arbre` comprendra une donnée privée, la racine qui est un nœud et le constructeur par défaut qui construit un arbre vide (sans nœud).

[On ne peut pas faire en faire grand chose pour l'instant, mais on va améliorer ces classes.]

Dans un **arbre de tri**, *on insère les éléments récursivement de la façon suivante : on a une liste d'entiers (non trié) sans doublon ; on va placer un nœud à l'arbre par entier ; le premier entier est placé dans la racine ; pour chaque entier suivant, si l'entier est strictement inférieur à la valeur de la racine, un nœud est ajouté à gauche à la bonne place, s'il est supérieur à la racine, il est ajouté à droite. Par exemple si on a l'arbre :*



et que l'on veut ajouter 12, on obtient l'arbre :



- 2^o) a) Ajouter la méthode `inserer` (un entier) à la classe `noeud` ayant un argument entier d : si d est strictement inférieur à la valeur du nœud et si le fils gauche est nul, on remplace ce fils gauche par une feuille de valeur d, si le fils gauche est non nul, on insère (récursivement) la valeur d à ce fils gauche ; si d est strictement supérieur à la valeur du nœud, on a un traitement analogue pour le fils droit.

b) Ajouter la méthode `inserer` (un nœud) à la classe `Arbre` dont l'argument est un entier : si la racine est nulle, la racine devient une feuille dont la valeur est cet entier ; sinon on insère cette valeur entière à la racine.

La **traversée en ordre** *d'un arbre binaire consiste à afficher les valeurs de ses nœuds récursivement de la façon suivante : si non vide traversée en ordre du fils gauche, racine, traversée en ordre du fils droit.*

On voit que, pour un arbre binaire de tri, on affiche ainsi les éléments dans l'ordre.

- 3^o) Ajouter une méthode `traversee` à la classe `Arbre`, qui affiche les éléments sur une ligne. Cette méthode fait appel à une méthode auxiliaire privée d'argument un nœud, qu'on appellera `assistant`.

- 4^o) Écrire une classe de test qui insère les valeurs 34, 23, 67, 12, 56, 3 et 5 à un arbre de tri et qui traverse cet arbre, c'est-à-dire qui trie la liste d'entiers ci-dessus.

Exercice 2.- (**Date**)

Écrire une applet Java qui demande une date sous la forme '19/12/01' et qui l'affiche sous la forme '19 décembre 2001'.

[Ne pas oublier le fichier html.]

DOCUMENTATION

La méthode :

String substring(int, int)

de la classe `String` permet de récupérer la sous-chaîne comprise entre les indices déterminés par le premier argument et le second argument moins un. Les indices commencent à 0.