Listes, Arbres

1) Pour faire un arbre, créer une classe Noeud qui poss'ede trois variables d'instance :
- filsDroit et filsGauche, deux références à des instances de Noeud ;
- valeur, un entier.

Cette classe possède un constructeur qui prend comme paramètre un entier et le range dans valeur, les autres variables ne référencent rien.

Elle possède une méthode ajoute prenant un entier en paramètre. Si cet entier est inférieur à valeur : si filsGauche ne référence rien, alors créer un nouveau Noeud (avec cet entier) et filsGauche référencera ce nouveau Noeud, sinon ajouter l'entier à filsGauche. Si l'entier est supérieur à valeur : faire de même avec filsDroit.

Cette classe possède aussi une méthode toString qui concatène le résultat de toString de filsGauche avec valeur et toString de filsDroit.

NB : si possible, aucune variable ne doit être déclarée public.

2) Créer une classe arbre qui contienne une variable racine de type Noeud, une m'ethode ajouter qui ajoute un entier dans l'arbre en bonne place et une méthode toString.

3) Créer une interface Comparable ayant une unique méthode inferieur prenant une autre instance de Comparable en paramètre.

Reprendre la question précédente en prenant la variable valeur de la classe Noeud du type Comparable.

Le tout formera un package Arbre.

4) Ecrire une sous-classe de String, StringComparable qui implémente l'interface Comparable.

Ecrire un programme qui range des chaines de caractères dans un arbre, puis les imprime dans l'ordre.