

Contrôle n°1

Durée : 1 heure. Aucun document autorisé.

Exercice 1 : (3 points) Qu'affiche le programme suivant ?

```
class A {
    private int x;
    private A a;
    public A(int x) {
        this.x = x;
        a = new A(x, 5);
    }
    public A(int n, int m) {
        x = m;
        a = this;
        System.out.print "[" + n + " ]";
    }
    public int f(int a) {
        System.out.print "+";
        if (this.a == this)
            return a;
        else return this.a.f(this.a) + a;
    }
    public int f (A a) {
        System.out.print "-";
        if (this.a == this)
            return a.x;
        else return this.a.f(a);
    }
    public static void main(String[] args) {
        A a = new A(7);
        System.out.println("A" + a.f(a.f(a)));
    }
}
```

Exercice 2 : (7 points)

1. Définir une classe `TableauTrie` permettant de représenter un tableau d'entiers dont les éléments sont toujours triés en ordre croissant. Définir un constructeur prenant en paramètre la taille maximale du tableau (dans la suite du sujet, on considérera que ce maximum est toujours respecté par les utilisateurs de la classe). Ajouter une méthode `public String toString()` affichant le contenu du tableau entre crochets.
2. Ajouter une méthode `public int getSize()` renvoyant le nombre d'éléments stockés dans le tableau, ainsi qu'une méthode `public int get(int index)` renvoyant la valeur stockée dans la case `index` du tableau (on considérera que cette dernière méthode est toujours appelée avec un `index` correct).
3. Ajouter une méthode `public void add(int n)` qui insère l'entier `n` dans le tableau trié. Utiliser la méthode `add` pour définir :
 - une méthode `public TableauTrie fusion(TableauTrie t)` qui renvoie un nouveau `TableauTrie` contenant la fusion des deux tableaux ;
 - un constructeur `public TableauTrie(int[] t)` qui initialise le `TableauTrie` en utilisant le tableau (non-trié) `t`.