

Le but de cet exercice est de définir des compteurs cycliques plus ou moins évolués. Un compteur cyclique est un compteur dont la valeur est régulièrement augmentée d'une unité jusqu'à ce qu'elle dépasse la valeur maximale et repasse à sa valeur initiale. Voici les méthodes que devra implémenter une classe modélisant un compteur cyclique :

- `boolean inc()` : ajoute la valeur 1 au compteur et retourne true si on est repassé à la valeur initiale (sinon retourne false).
- `void affiche()` : affiche la valeur du compteur.

1. *Ecrivez la classe `CompteurSimple` qui implémente les méthodes ci-dessus et définit notamment un constructeur prenant 3 paramètres : la valeur entière initiale du compteur, la valeur maximale et le nom (de type `String`) de l'unité dont le compteur contient la valeur. Exemple d'utilisation :*

```
CompteurSimple trimestre = new CompteurSimple(1, 4, "Trimestre");
trimestre.affiche(); // affiche "Trimestre: 1"
trimestre.inc(); trimestre.inc(); trimestre.inc();
trimestre.affiche(); // affiche "Trimestre: 4"
trimestre.inc();
trimestre.affiche(); // affiche "Trimestre: 1"
```

2. Maintenant, on veut pouvoir créer des compteurs plus complexes qui mettent en jeu plusieurs valeurs dont chacune a une unité différente. Par exemple, si on veut créer une horloge, on a besoin d'une valeur pour les minutes et une valeur pour les heures. Ces compteurs seront donc composés d'une suite de compteurs simples. Incrémenter un compteur composé, c'est incrémenter son compteur simple qui contient la valeur de la plus petite unité (ex: les minutes) et propager l'incrémentation au reste des compteurs dont il est composé (ex: les heures) si la valeur a dépassé le maximum et a été réinitialisée.

Ecrivez la classe `CompteurCompose` qui implémente les mêmes méthodes mais avec deux constructeurs différents, l'un prenant un seul paramètre de type `CompteurSimple` et l'autre prenant en plus un paramètre de type `CompteurCompose`, pour permettre par exemple la création et l'utilisation d'un compteur composé de la manière suivante :

```
CompteurCompose c;
c = new CompteurCompose(new CompteurSimple(0, 59, "Minute"),
                        new CompteurCompose(new CompteurSimple(0, 23, "Heure"),
                        new CompteurCompose(new CompteurSimple(1, 365,
                        "Jour"))));
c.affiche(); // affiche "Jour: 1  Heure: 0  Minute: 0"
for(i = 1; i < 10000; i++)
    c.inc();
c.affiche(); // affiche "Jour: 7  Heure: 22  Minute: 39"
```

3. *Ajoutez une méthode de copie et de comparaison aux classes `CompteurSimple` et `CompteurCompose`. Testez-les dans un programme.*