

Laboratoire 4: Types primitifs et classes

Durée du laboratoire: 2 périodes.

1. Tri de tableau d'entiers

Ecrire un programme permettant de récupérer des valeurs entières fournies en argument de la ligne de commande dans un tableau, de trier ce tableau en appliquant la méthode des échanges et d'afficher son contenu.

La transformation d'un argument de la ligne de commande (un objet `String`) en valeur entière sera effectuée en parcourant les caractères qui le composent (c.f. méthodes de la classe `String` dans l'API et non pas en utilisant la méthode `int parseInt(String s)` de la classe `Integer`). Attention aux signes `+` et `-` qui peuvent précéder un nombre.

2. Classe `Int`

Définir une classe `Int` encapsulant le type primitif `int`. Elle devra fournir des méthodes permettant de:

- Construire des instances de cette classe,
- Consulter la valeur entière encapsulée dans l'objet courant,
- Modifier la valeur entière encapsulée dans l'objet courant (contrairement à la classe `Integer` de l'API),
- Rendre une représentation de l'objet courant sous la forme d'un objet `String`.

3. Tri de tableau d'objets

Modifier le programme de l'exercice 1 en utilisant non plus un tableau d'entiers mais un tableau d'objets `Int`. Implanter la méthode des échanges de trois manières différentes:

1. En définissant une méthode permettant d'échanger deux éléments d'un tableau d'objets `Int`.
2. En définissant une méthode permettant d'échanger la valeur de deux objets `Int`.
3. En définissant une méthode dans la classe `Int` permettant d'échanger la valeur de l'objet courant avec celle d'un autre objet `Int`.

Remarque

Les méthodes définies au points 1 et 2 sont fondamentalement différentes de celle définie au point 3: elles ne sont pas invoquées sur un objet donné. Dès lors, elles doivent être définies comme des méthodes *de classe* (au moyen du mot clef `static`).