

# TP 5: Boucles et tableaux

Informatique Fondamentale (IF121)

16 Novembre 2004

**Rappel.** Sauf si l'énoncé le précise explicitement, une méthode n'est pas censée lire ou afficher quoi que ce soit.

## 1 Suites numériques : la suite

**Exercice 1 :** *Puissance d'un nombre*

Écrire une méthode qui calcule la puissance  $n^{\text{ème}}$  d'un nombre, où  $n$  est un entier positif. Par exemple, `puissance(2.5, 3)` vaut 15,625.

Modifier la méthode précédente pour traiter le cas où  $n$  est un entier quelconque. Par exemple, `puissance(2.5, -3)` vaut 0,064; `puissance(2.5, 0)` vaut 1.

**Exercice 2 :** *Une série*

Écrire une méthode qui calcule la somme des carrés des  $n$  premiers entiers *impairs*. Par exemple, `sommeCarresImpairs(5)` =  $1^2 + 3^2 + 5^2 + 7^2 + 9^2 = 165$ .

**Exercice 3 :** *Une suite*

On définit la suite  $(x_n)_{n \in \mathbb{N}}$  de la manière suivante :  $x_0 = a$  et  $x_{n+1} = 4x_n(1 - x_n)$ . Écrire une méthode qui calcule  $x_n$  en fonction de  $a$  et de  $n$ . Modifier celle-ci pour qu'elle affiche les valeurs successives  $x_0, x_1, \dots, x_n$  de la suite.

## 2 Affichage

**Exercice 4 :** *Compter*

Écrire un programme qui lit un entier  $n$ , puis qui affiche tous les entiers de 1 à  $n$  (un par ligne).

**Exercice 5 :** *Un rectangle*

(a) Écrire une méthode `repetePrint` qui prend comme argument un entier  $n$  et un caractère, et qui affiche  $n$  fois de suite (sans passer à la ligne) le caractère en question.

(b) À l'aide de la méthode `repetePrint`, écrire un programme qui affiche (sous forme de texte) un rectangle de côtés donnés. On utilisera les caractères -, | et +. Par exemple :

```
> java Rectangle
Largeur : 10
Hauteur : 5
+-----+
|         |
|         |
|         |
+-----+
```

### 3 Traitement de données

#### Exercice 6 : *La ligne la plus longue*

Écrire un programme qui lit un entier  $n$ , puis qui lit  $n$  lignes, et affiche la ligne la plus longue. Exemple :

```
> java LigneLaPlusLongue
Combien de lignes ? 5
[1] Ceci
[2] est
[3] un
[4] essai
[5] ...
Ligne la plus longue : essai
```

Qu'affiche votre programme s'il y a des ex-aequos ?

#### Exercice 7 : *Compter les voyelles*

Écrire un programme qui lit une ligne et affiche le nombre de voyelles qu'elle contient.

#### Exercice 8 : *Arguments d'un programme*

L'argument de la méthode `main` est le tableau des arguments du programme. Par exemple, si on exécute le programme `Toto` par

```
java Toto bonjour tout le monde
```

la méthode `main` de `Toto` reçoit un tableau à 4 éléments : "bonjour", "tout", "le" et "monde".

Écrire un programme `Echo` qui affiche ses arguments, séparés par des espaces.

### 4 Tableaux de nombres

#### Exercice 9 : *Lecture d'un tableau de nombres*

Écrire une méthode `litTableau` qui lit un entier  $n$ , puis qui lit un  $n$  nombres réels qu'elle place dans un tableau, et qui renvoie ce tableau.

#### Exercice 10 : *Concaténation*

Écrire une méthode `concatene` qui prend comme arguments deux tableaux `t1` et `t2` et qui construit et renvoie un tableau `t` contenant les éléments de `t1` suivis des éléments de `t2`.

#### Exercice 11 : *Vérification de tri*

Écrire une méthode qui prend comme argument un tableau de nombres et qui vérifie si ce tableau est trié en ordre croissant.

#### Exercice 12 : *Fusion de deux tableaux triés*

Écrire une méthode qui prend comme arguments deux tableaux de nombres `t1` et `t2`, chacun trié en ordre croissant, et qui construit un nouveau tableau `t` trié contenant les mêmes éléments. Si un nombre apparaît  $k_1$  fois dans `t1` et  $k_2$  fois dans `t2`, il doit apparaître  $k_1 + k_2$  fois dans `t`. Par exemple, la fusion de `(3, 3, 6, 7)` avec `(1, 2, 3, 5, 6)` est `(1, 2, 3, 3, 3, 5, 6, 6, 7)`.

#### Exercice 13 : *Norme du sup*

On appelle « norme du sup » du vecteur  $\vec{x}$  de coordonnées  $(x_0, \dots, x_{n-1})$  le nombre réel positif donné par

$$\|\vec{x}\|_\infty = \max\{|x_0|, |x_1|, \dots, |x_n|\}$$

Écrire une méthode qui calcule la norme du sup d'un vecteur.