

TP 6 : tableaux et boucles **for**

Informatique Fondamentale (IF1)

Semaine du 27 novembre 2007

1 Arguments de ligne de commande

L'argument de la fonction `main` est un tableau qui contient les arguments du programme. Par exemple, si on exécute le programme `Toto` en tapant :

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

la fonction `main` de la classe `Toto` reçoit en paramètre un tableau à quatre éléments (numérotés de 0 à 3) : `"bonjour"`, `"tout"`, `"le"` et `"monde"`.

1. Écrivez un programme `Echo` qui affiche ses arguments, séparés par des espaces. Par exemple :

```
$ java Echo ala ma kota  
ala ma kota
```

N'oubliez pas d'aller à la ligne après le dernier argument.

2 Vecteurs

Dans cette partie, on représente le vecteur de coordonnées (x_0, \dots, x_{n-1}) , d'un espace vectoriel de dimension n , par le tableau de taille n dont les éléments sont x_0, \dots, x_{n-1} . Vous répondrez aux questions de cette partie dans une classe `Vect`.

2. Écrivez une fonction `produitScalaire`, qui calcule le produit scalaire des deux vecteurs de même dimension `vect1` et `vect2` :

```
public static double produitScalaire(double[] vect1, double[] vect2)
```

3. Écrivez une fonction `main`, qui vous permette de vérifier que le produit scalaire des vecteurs $(3, 4)$ et $(4, 5)$ est bien 32.

La norme $|\mathbf{x}|$ d'un vecteur $\mathbf{x} = (x_0, \dots, x_{n-1})$ peut-être définie par la racine carrée du produit scalaire de \mathbf{x} avec lui-même :

$$|\mathbf{x}| = \sqrt{\mathbf{x} \cdot \mathbf{x}} = \sqrt{\sum_{i=0}^{n-1} x_i^2}.$$

4. Écrivez une fonction `norme`, qui, à l'aide de la fonction `produitScalaire`, calcule la norme d'un vecteur `vect` :

```
public static double norme(double[] vect)
```

5. Complétez la fonction `main` pour qu'elle permette de vérifier que la norme du vecteur $(3, 4)$ est 5.

6. Écrivez une fonction `litVecteurDouble`, qui demande à l'utilisateur un entier n , crée un tableau a de taille n , puis demande n nombres réels qu'elle stocke dans le tableau a . Elle termine en renvoyant le tableau a .

```
public static double[] litVecteurDouble()
```

7. Modifiez la fonction `main` pour qu'elle affiche la norme d'un vecteur qu'elle a lu à l'aide de la fonction `litVecteurDouble`.

3 Moyenne Harmonique

On appelle *moyenne harmonique* H d'une famille de nombres (x_0, \dots, x_{n-1}) la valeur :

$$\begin{aligned} H &= \frac{n}{\sum_{i=0}^{n-1} \frac{1}{x_i}} && \text{si tous les } x_i \text{ sont non-nuls} \\ H &= 0 && \text{sinon} \end{aligned}$$

8. Écrivez un programme qui lit une famille de n nombres, puis affiche leur moyenne harmonique.

Vous pourrez bien-sûr vous servir de la fonction `litVecteurDouble` définie dans la classe `Vect`.

Rappel : Pour appeler une fonction définie dans une autre classe, il faut préfixer le nom de la fonction par le nom de la classe où elle est définie. Par exemple :

```
double[] v = Vect.litVecteurDouble();
```

4 Tableaux triés

Vous répondrez aux questions de cette section dans une classe `Tri`.

9. Écrivez une fonction `litVecteurInt`, qui demande à l'utilisateur un vecteur d'entiers. La programmation par couper/coller est bien sûr une technique acceptable dans cet exercice.

10. Écrivez une fonction `estCroissant` qui prend comme argument un tableau d'entiers et renvoie `true` si ce tableau est trié en ordre croissant et `false` sinon.

11. Écrivez une fonction `main` qui lit un tableau d'entiers et affiche s'il est trié en ordre croissant.

5 Recherche d'un entier dans un tableau

Vous répondrez aux questions de cette section dans une classe `Recherche`.

12. Écrivez une fonction :

```
int rechercheLineaire(int valeur, int[] a)
```

qui retourne le plus petit i tel que $a[i] = \text{valeur}$, et -1 si aucun des éléments de `a` ne vaut `valeur`.

13. Écrivez une fonction `main` qui lit un entier `valeur` et un tableau d'entiers `a`, puis recherche et affiche un indice i tel que $a[i] = \text{valeur}$. Elle affichera -1 si aucun des éléments de `a` ne vaut `valeur`.