TP 3 : premières boucles

Informatique Fondamentale (IF121)

3 novembre 2004

1 Suites numériques

Exercice 1 : *Puissance d'un nombre*

Écrire un programme qui demande un nombre flottant x ainsi qu'un entier positif n à l'utilisateur et qui calcule la puissance $n^{\text{ème}}$ de x.

Écrire une autre version de ce programme pour traiter le cas où n est un entier quelconque (éventuellement négatif ou nul). Par exemple, 2.5^{-3} vaut $0,064$; 2.5^0 vaut 1.

Exercice 2 : *Factorielle*

On rappelle que la fonction factorielle est définie sur les entiers positifs de la façon suivante :

$$\begin{aligned} \text{factorielle}(0) &= 1 \\ \text{factorielle}(n) &= n \times \text{factorielle}(n-1) \qquad \text{si } n \geqslant 1 \end{aligned}$$

Écrire un programme qui calcule la factorielle d'un entier fourni par l'utilisateur.

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 un programme qui calcule x_n en fonction de $a \in \mathbb{R}$ et de $n \in \mathbb{N}$, fournis par l'utilisateur.

Exercice 4 : *Approximation de $\sqrt{x}$*

Étant donné un réel strictement positif a, on définit la suite réelle $(x_n)_{n\in\mathbb{N}}$ de la manière suivante :

$$x_0 = a \qquad \text{et} \qquad x_{i+1} = \frac{x_i^2 + a}{2x_i}$$

On admet que cette suite converge vers $\sqrt{a}$.

Écrire un programme qui lit un réel positif a et un entier n et qui renvoie une valeur approchée `b` de $\sqrt{a}$ en utilisant l'approximation x_n. Afficher aussi `b*b` pour vérifier.

Exercice 5 : *Une série*

Écrire un programme qui demande un entier positif n à l'utilisateur et qui calcule la somme des carrés des n premiers entiers. Par exemple, si $n = 5$, le résultat est $1^2 + 2^2 + 3^2 + 4^2 + 5^2 = 55$.

Exercice 6 : *Une autre série*

Écrire un programme qui demande un entier positif n à l'utilisateur et qui calcule la somme des carrés des n premiers entiers *impairs*. Par exemple, si $n = 5$, le résultat est $1^2 + 3^2 + 5^2 + 7^2 + 9^2 = 165$.

2 Suites numériques (encore)

Exercice 7 : *Deux suites*

On définit une suite double :

$$\begin{aligned} u_0 &= 1 & u_{n+1} &= (u_n + v_n)/2 \\ v_0 &= 2 & v_{n+1} &= \sqrt{u_{n+1} v_n} \end{aligned}$$

On admet que les suites (u_n) et (v_n) sont adjacentes de limite $\sqrt{27}/\pi$. Écrire un programme qui lit un entier n et affiche l'approximation du nombre π obtenue à partir de v_n.

Exercice 8 : *Suite de Fibonacci*

On définit la suite de Fibonacci de la manière suivante :

$$F_0 = F_1 = 1$$
$$F_{n+2} = F_n + F_{n+1}$$

Écrire un programme qui demande un entier positif ou nul n à l'utilisateur et qui calcule le $n^{\text{ème}}$ nombre de Fibonacci (c'est-à-dire F_n).

Exercice 9 : *Somme des puissances $k^{èmes}$*

Écrire un programme qui, étant donnés deux entiers k et n fournis par l'utilisateur, calcule la somme des puissances $k^{\text{èmes}}$ des n premiers entiers. Par exemple, si $k = 3$ et $n = 5$, le résultat est $1^3 + 2^3 + 3^3 + 4^3 + 5^3 = 225$. (On pensera à réutiliser le code de l'exercice 1.)

3 Affichage

Exercice 10 : *Compter*

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

Exercice 11 : *Un rectangle*

(a) Écrire un programme qui demande un entier n et un caractère, et qui l'affiche n fois de suite (sans passer à la ligne).

(b) Écrire un programme qui demande deux entiers positifs k et n à l'utilisateur et qui affiche (sous forme de texte) un rectangle de longueur k et de largeur n. On utilisera les caractères `-`, `|` et `+`. Par exemple :

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

4 Traitement de données

Exercice 12 : *Calcul de moyenne*

Écrire un programme qui lit un entier n, puis qui lit n nombres, et affiche la moyenne de ces nombres. Exemple :

```
> java Moyenne
Combien de nombres ? 4
Nombre 1 : 3
Nombre 2 : 5
Nombre 3 : 1
Nombre 4 : 55
La moyenne est 16.
```

Exercice 13 : *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 ?

5 Manipulation de texte

Exercice 14 : *Somme des chiffres*

La méthode `Deug.intToString` permet d'obtenir la représentation décimale d'un nombre entier. Par exemple, `Deug.intToString(42)` est la chaîne `"42"`.

Étant donné un caractère `c` représentant un chiffre, on peut obtenir le nombre entier correspondant par `c - '0'`. Par exemple, l'extrait de programme suivant affiche `6` :

```
char c = '3';
int n = 2 * (c - '3');
Deug.println(n);
```

Écrire un programme qui demande un entier positif n et qui calcule la somme de ses chiffres. Par exemple, si $n = 1789$, le résultat est 25.

Exercice 15 : *Compter les voyelles*

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

Exercice 16 : *Palindrome*

Un texte est un palindrome de lettres si c'est une succession de lettres symétrique. Par exemple, « ici » est un palindrome de lettres tout comme « engagelejeuquejelegagne ».

Écrire un programme qui demande une succession de lettres à l'utilisateur et lui dit si ce texte est un palindrome ou non.

6 Dessin

Exercice 17 : *Polygone régulier*

Écrire un programme qui demande un entier n à l'utilisateur et qui dessine un polygone régulier à n côtés.