

PROGRAMMATION 1

TD1

Notions à acquérir

Déclaration des variables et définition des types simples (entiers, réels, booléens et caractères)
Instructions simples : affectation
Algorithmes séquentiels
Instructions itératives : Boucle **pour**
Boucle **tant que**
Spécifications et commentaires pertinents
Trace des algorithmes

Exercices sur l'affectation

Exercice 1

Effectuer la trace du programme suivant :

variables

Nbre , X, Valeur : entier

début

Nbre	←	10
Valeur	←	8
X	←	Valeur – Nbre
Valeur	←	Valeur * 5
X	←	Valeur - Nbre
X	←	X - Nbre

fin

Exercice 2

a) Quelles seront les valeurs des variables A et B après exécution du programme suivant :

variables

A, B : entier

début

A	←	10
B	←	3
A	←	B
B	←	A

fin

- b) Les deux dernières instructions permettent-elles d'échanger les deux valeurs de A et B ?
c) Si l'on inverse l'ordre de ces deux dernières instructions, cela change-t-il quelque chose ?
d) Remplacer les deux dernières instructions par une suite d'instructions permettant d'échanger les deux valeurs de A et B indépendamment des valeurs données à A et B (10 et 3).

Exercice 3

Dans l'algorithme qui suit, discuter la validité et la pertinence de chaque affectation :

variables

a, b, c, d : entier

c1, c2 : caractère

u, v : booléen

début

```
a      ← 10
d      ← a
c      ← b + c
c1     ← 'a'
c2     ← 'b'
u      ← c1 == c2
v      ← vrai
u      ← u ou v
v      ← a + v
a      ← c1 / 4
a      ← d / 5
```

fin

Exercice 4

Soient les deux types :

type T_COULEUR = (bleu, rouge, vert, jaune)

type T_FEUX = (rouge, orange, vert)

et soient les variables suivantes

C1, C2 : T_COULEUR

F1, F5 : T_FEUX

Discuter de la validité et de la pertinence des instructions suivantes :

début

```
C1      ← rouge
C2      ← orange
F1      ← C1
F1      ← rouge
F5      ← F1
F5      ← vert
C1      ← vert + rouge
```

fin

Exercices sur les boucles pour et tantque

Exercice 5

Ecrire un algorithme qui permet de saisir au clavier un nombre entier, puis qui affiche à l'écran les dix nombres suivants.

Exercice 6

- a) Ecrire un programme qui demande à l'utilisateur de saisir un entier N, puis de saisir N entiers. Le programme affiche alors la somme de ces N entiers.
- b) Ecrire un programme qui demande à l'utilisateur de saisir des entiers jusqu'à ce que l'utilisateur saisisse la valeur 0. Le programme affiche alors le nombre d'entiers non nuls qui ont été saisis et leur somme.

Exercice 7

Soient les trois algorithmes A1, A2 et A3 suivants, où X et B sont deux variables de type entier :

A1 :

```
B ← 8
saisir ( X )
tantque ( B < X ) faire
    B ← B + 3
    X ← B - 2
fintantque
```

A2 :

```
B ← 8
saisir ( X )
tantque ( B < X ) faire
    X ← B - 2
    B ← B + 3
fintantque
```

A3 :

```
B ← 8
saisir ( X )
tantque ( B < 10 ) faire
    X ← B - 2
    B ← B + 3
fintantque
```

- a) Quelle est la valeur de X après l'exécution de chacun des algorithmes, si pour X on saisit 6, puis 15 (en relançant le programme) et enfin 33?
- b) A1, A2 et A3 (ou deux d'entre eux) sont ils des algorithmes équivalents, c'est-à-dire qui donnent les mêmes valeurs finales de B et de X dans tous les cas?
- c) Quelle est la valeur de X après l'exécution de chacun des algorithmes, si pour X on saisit un entier A (donner la réponse en fonction de A)?

Exercice 8

- a) Ecrire un algorithme qui
demande à l'utilisateur d'entrer un nombre N de personnes
demande à l'utilisateur d'entrer les ages de ces N personnes (des entiers)
affiche la moyenne de ces ages

Que se passe-t-il si N=0?

- b) Compléter l'algorithme précédent en demandant à l'utilisateur de ressaisir la valeur N jusqu'à ce qu'elle soit différente de 0 avant de lui demander de saisir les ages.

c) Ecrire un algorithme qui répète celui de la question a) jusqu'à ce que l'utilisateur saisisse la valeur 0 pour N.

Exercice 9

a) Soit la suite U_n définie par :

$$\begin{aligned} U_1 &= 1 \\ U_n &= U_{n-1} + n \quad \text{pour } n > 1 \end{aligned}$$

Ecrire un algorithme qui demande à l'utilisateur de saisir un entier n (que l'on suppose strictement positif) et affiche U_n .

b) Même question pour la suite U_n définie par :

$$\begin{aligned} U_1 &= 1 \\ U_n &= U_{n-1} + \cos(n) \quad \text{pour } n > 1 \end{aligned}$$

c) Même question pour la suite U_n (appelée suite de Fibonacci) définie par :

$$\begin{aligned} U_1 &= U_2 = 1 \\ U_n &= U_{n-1} + U_{n-2} \quad \text{pour } n > 2 \end{aligned}$$

Exercice 10

Ecrire un programme qui demande à l'utilisateur de saisir deux entiers n et p (on suppose n et p strictement positifs), et qui affiche le reste et le quotient de la division euclidienne de n par p. L'algorithme ne doit utiliser que les opérateurs + ou -.

Exercice 11

Ecrire un algorithme qui permette de calculer une valeur approchée de $\pi/4$ à EPSILON près, sachant que $\pi/4$ est la limite de la série :

$$S = 1 - 1/3 + 1/5 - 1/7 + 1/9 - \dots$$

On admettra qu'il suffit d'arrêter l'itération lorsque le terme additionnel devient inférieur à EPSILON.

Exercice 12

Montrer que toute boucle pour peut être réécrite avec une boucle tantque. Réécrire les boucles

```
pour i dans a..b faire  
    <suite d'instructions>  
finpour »
```

et

```
pour i dans a..b en ordre inverse faire  
    <suite d'instructions>  
finpour
```

où a et b sont des entiers, sous forme de boucles tantque.