

# TD 1: Algorithmique fonctionnelle

## Concepts de base

Nga Nguyen - Stefan Bornhofen - Peio Loubière

### 1 Variables

Quelle est, après chaque instruction la valeur de chacune des Variables suivantes :

```
1 Programme
2 Variables
3   a, b, c, d : Entier
4   e, f : Booleen
5 Début
6   a ← 7
7   b ← 12
8   c ← b - a
9   c ← c * 3
10  d ← a
11  a ← b
12  f ← faux
13  e ← non f
14  e ← e ou (b=4)
15  f ← (b=12) et (c=2)
16 Fin
```

*Solution...*

|   | 1 | 2  | 3  | 4  | 5  | 6  | 7    | 8    | 9    | 10   |
|---|---|----|----|----|----|----|------|------|------|------|
| a | 7 | 7  | 7  | 7  | 7  | 12 | 12   | 12   | 12   | 12   |
| b |   | 12 | 12 | 12 | 12 | 12 | 12   | 12   | 12   | 12   |
| c |   |    | 5  | 15 | 15 | 15 | 15   | 15   | 15   | 15   |
| d |   |    |    |    | 7  | 7  | 7    | 7    | 7    | 7    |
| e |   |    |    |    |    |    |      | vrai | vrai | vrai |
| f |   |    |    |    |    |    | faux | faux | faux | faux |

## 2 Lecture / Écriture

Écrivez un programme qui convertit des euros en dollars : il demande un nombre réel, censé représenter une somme en euro, et qui affiche le montant correspondant en dollars.

Le taux de change est d'environ 1 euro = 1,4676 dollars.

*Solution...*

```
1 Programme
2 Variables
3   montant : Reel
4 Début
5   Ecrire (" Entrez un montant à convertir : ")
6   Lire(montant)
7   montant ← 1,4676*montant
8   Ecrire (" La somme en dollars est " & montant)
9 Fin
```

## 3 Instruction conditionnelle

### 3.1 Convertisseur

Améliorez le programme ci-dessus en proposant, avant la saisie du montant, une conversion euro → dollar ou bien dollar → euro.

*Solution...*

```
1 Programme
2 Variables
3   conv : Entier
4   montant : Reel
5 Début
6   Ecrire (" Voulez-vous convertir : ")
7   Ecrire (" 1 : dollars -> euro")
8   écrire (" 2 : euro -> dollars")
9   Lire(conv)
10  Ecrire (" Entrez un montant à convertir : ")
11  Lire(montant)
12  Si conv=1 Alors
13    montant ← montant/1,4676
14  Sinon
15    montant ← montant*1,4676
16  FinSi
17  Ecrire " La somme convertie est " & montant
18 Fin
```

### 3.2 Absolutement !

Écrire un programme qui demande un nombre réel à l'utilisateur et calcule sa valeur absolue.

*Solution...*

```
1 Programme Absolutemen
2 Variables
3   val : Entier
4 Début
5 Ecrire ("Entrez un entier : ")
6 Lire(val)
7 Si val<0 Alors
8   val ← -1*val
9 FinSi
10 Ecrire ("La valeur absolue est " & val)
11 Fin
```

### 3.3 Équation du second degré

Écrire un programme qui résout, dans  $\mathbb{R}$ , une équation du second degré. Il demande de saisir les coefficients a, b, c de l'équation,  $a \neq 0$ , calcule et affiche la solution.

*Solution...*

```
1
2 Programme resolEq2D
3 Variables :
4   a, b, c : Entier
5   Δ : Reel
6   res : Chaine
7 Debut
8   Ecrire("entrez le premier coefficient , non nul")
9   Lire(a)
10  Si (a=0) Alors
11    Ecrire("Ce n'est pas une équation du second degré")
12  Sinon
13    Ecrire("entrez le deuxième coefficient")
14    Lire(b)
15    Ecrire("entrez le troisième coefficient")
16    Lire(c)
17    Δ = b2 - 4*a*c
18    Si Δ < 0 Alors
19      res = "L'équation n'admet pas de solution dans  $\mathbb{R}$ "
20    Sinon Si Δ = 0 Alors
```

```

21     res = "L'équation n'admet une unique solution dans
22     ℚ, x=" &  $\frac{-b}{2*a}$ 
23     Si  $\Delta < 0$ 
24     res = "L'équation n'admet 2 solutions dans ℚ, x1 =
25     " &  $\frac{-b-\sqrt{\Delta}}{2*a}$  & "x2=" &  $\frac{-b+\sqrt{\Delta}}{2*a}$ 
26     FinSi
27     FinSi
28     Ecrire(res)
29 Fin

```

### 3.4 Divination...

Cet algorithme est destiné à prédire l'avenir, et il doit être infaillible! Il lira au clavier l'heure et les minutes, et il affichera l'heure qu'il sera une minute plus tard. Par exemple, Si l'utilisateur tape 21 puis 32, l'algorithme doit répondre : "Dans une minute, il sera 21 heure(s) 33". On suppose que l'utilisateur entre une heure valide. Pas besoin donc de la vérifier.

*Solution...*

```

1  Programme Divination
2  Variables
3    h, m : Entier
4  Début
5    Ecrire ("Entrez les heures , puis les minutes : ")
6    Lire(h, m)
7    m ← m + 1
8    Si m = 60 Alors
9      m ← 0
10     h ← h + 1
11   FinSi
12   Si h = 24 Alors
13     h ← 0
14   FinSi
15   Ecrire ("Dans une minute il sera " & h & "heure(s) " & m
16     & "minute(s)")
17 Fin

```

### 3.5 Congés

Dans une entreprise, le calcul des jours de congés payés s'effectue de la manière suivante :

- Si une personne est entrée dans l'entreprise depuis moins d'un an, elle a droit à deux jours de congés par mois de présence Sinon à 28 jours au moins;

- Si c'est un cadre et s'il est âgé d'au moins 35 ans et Si son ancienneté est supérieure à 3 ans, il lui est accordé 2 jours supplémentaires ;
- s'il âgé d'au moins 45 ans et si son ancienneté est supérieure à 5 ans, il lui est accordé 4 jours supplémentaires, en plus des 2 accordés pour plus de 35 ans.

Écrire un algorithme qui calcule et affiche le nombre de jours de congés à partir de l'âge, de l'ancienneté et de l'appartenance au collège cadre d'un employé.

*Solution...*

```

1  Programme CalculConge
2  Variables :
3    age , anc , nbJourDeConge : Entier
4    rep : Chaîne
5    cadre : Booleen
6  Début
7    Ecrire("entrez l'âge du salarié")
8    Lire(age)
9    Ecrire("entrez son ancienneté")
10   Lire(anc)
11   Ecrire("est-ce un cadre (O/N)")
12   Lire(rep)
13   //initialisation des variables
14   cadre ← rep="O"
15   nbJourDeConge ← 0
16   //calcul
17   Si (anciennete <12) Alors
18     nbJourDeConge ← anciennete*2
19   Sinon
20     nbJourDeConge ← 28
21   FinSi
22   Si cadre Alors
23     Si (age ≥ 35) et (anciennete ≥ 3*12) Alors
24       nbJourDeConge ← nbJourDeConge+2
25     FinSi
26     Si (age ≥ 45) et (anciennete ≥ 5*12) Alors
27       nbJourDeConge ← nbJourDeConge+4
28     FinSi
29   FinSi
30   Ecrire (" le nombre de congés accumulés est " &
31     nbJourDeConge)
Fin

```