SOMMAIRE

1. Analyse d’un problème 2
2. Arbre programmatique 2
3. Représentation d'une fonction 3
4. VBA 4
5. Notion de variables 5
6. String 7
7. Alternative 8
8. Booléen 12
9. Sous routine 15
10. Itérative 16
11. Tableaux 20
12. Pour element dans liste 23
13. Chaîne de caractères 24

ARBRE PROGRAMMATIQUE ET PROGRAMMATION VBA

1. ANALYSE D'UN PROBLEME

L'analyse d'un problème complexe consiste en sa décomposition en une suite de problèmes indépendants moins complexes que l'on pourra décomposer à leur tour.

Exemple :

cuire un œuf à la coque :

- mettre l'œuf dans l'eau bouillante :
  - prendre une casserole
  - mettre de l'eau dans la casserole :
    - prendre une casserole
    - mettre la casserole sous un robinet
    - ouvrir le robinet
    - fermer le robinet lorsque la casserole est pleine
  - faire chauffer l'eau
  - mettre l'œuf dans l'eau
- attendre trois minutes :
  - prendre un sablier
  - attendre qu'il se vide
- retirer l'œuf

Nous verrons que tout problème peut se résoudre en une décomposition de séquences, de conditions et d'itérations.

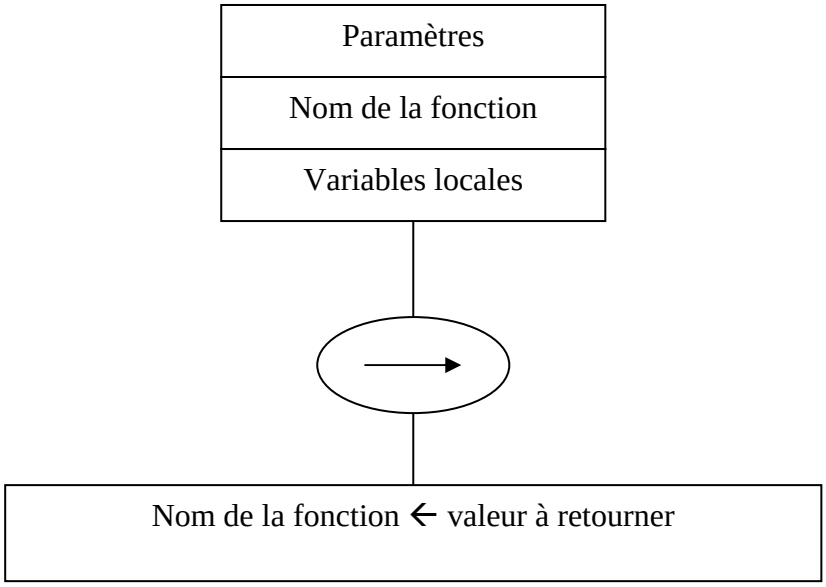
Séquence : suite d'actions réalisées les unes après les autres.
Condition : réalisation d'une action ou d'une autre selon une condition.
Itération : exécution des mêmes actions autant de fois que nécessaire.

2. ARBRE PROGRAMMATIQUE

Afin de rendre la représentation et la compréhension, d'un algorithme, les plus simples et les plus rapides possibles, différents outils ont été développés. L'arbre programmatique est un de ces outils. Il consiste en la représentation de l'algorithme à l'aide de symboles graphiques. L'un des avantages de cet outil est sa simplicité dans sa transformation en langage de programmation.

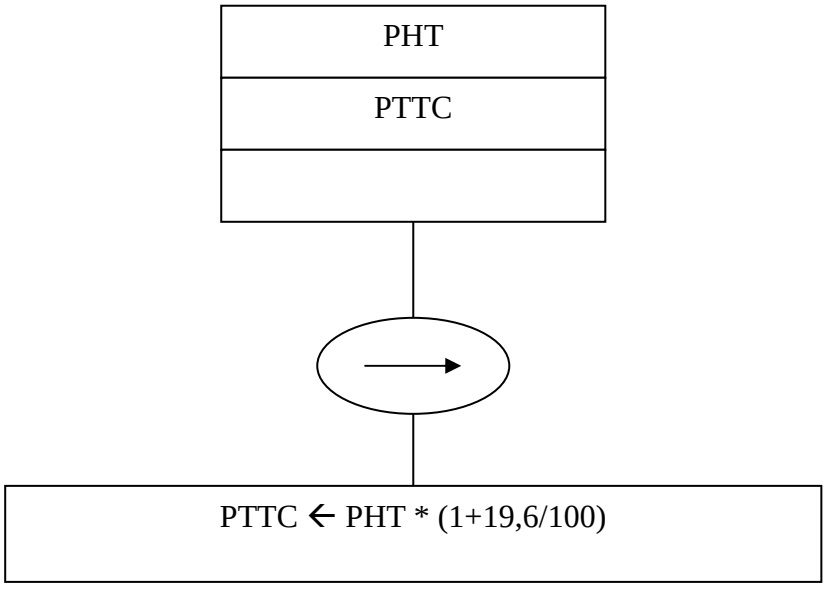
3. REPRESENTATION D'UNE FONCTION

Le tableur Excel comporte plusieurs fonctions prédéfinies. Pour utiliser une de ces fonctions il suffit d'écrire : =cos(10) et on obtient un résultat. Cos étant le nom de la fonction, 10 la valeur du paramètre. Dans la suite nous montrons comment on représente une telle fonction sous forme d'arbre programmatique.



3.1. Exemple

Retourne le prix TTC à partir du prix hors taxe



Ce qui donne :

PTTC(10) = 11,96
PTTC(20) = 23,92

3.2. Application

1) **Retourne le périmètre d'un cercle connaissant son rayon. (PI = 3,14)**
2) **Retourne la surface d'un cercle connaissant son diamètre.**
3) **Retourne une vitesse connaissant le temps et la distance.**

4. VBA

Pour valider nos algorithmes, nous utiliserons le langage Visual Basic dans le logiciel EXCEL.

4.1. Editeur

4.2. Fonction

```
Function nom_fonction(paramètres)
     nom_fonction = valeur retournée
End Function
```

Il est conseillé de faire précéder le nom de ses fonctions par un signe distinctif et personnel. Afin de ne pas retomber sur des noms existants.

Par exemple de faire précéder ses fonctions par ses initiales.

4.3. Exemple

```
Function vv_pttc(pht)
  vv_pttc = pht * (1 + 19.6 / 100)
End Function
```

Nous venons de créer une nouvelle fonction **VV_PTTC** qui est désormais accessible dans excel. Pour tester notre fonction, il suffit de se positionner dans un feuille de notre classeur et de taper dans une cellule **=VV_PTTC(10)** et la cellule affiche le résultat du calcul.

4.4. Application

1) **Retourne le périmètre d'un cercle connaissant son rayon. (PI = 3,14)**
2) **Retourne la surface d'un cercle connaissant son diamètre.**
3) **Retourne une vitesse connaissant le temps et la distance.**

5. NOTION DE VARIABLES

Pour résoudre un problème il nous faut conserver des informations (Age, nom, nombre…). Ces données sont appelées variables.

Un peu comme une lettre postal, une variable englobe plusieurs informations :

	son adresse			&1000		&3000
	son nom			i		c
Variable :	son contenu	ex :	variable i :	5	variable c :	‘A’
	son type			entier		caractère

5.1. L’adresse

L’adresse correspond à l’emplacement physique de la variable dans la mémoire de l’ordinateur (adresse de la personne à qui l'on écrit)

5.2. Le nom

Plutôt que d’appeler une variable par son adresse, on lui donne un nom afin de simplifier son utilisation (nom du destinataire).

5.3. Le contenu

Le contenu d’une variable peut être un nombre, un caractère, un mot, un tableau…(lettre se trouvant dans l'enveloppe)

5.4. Le type

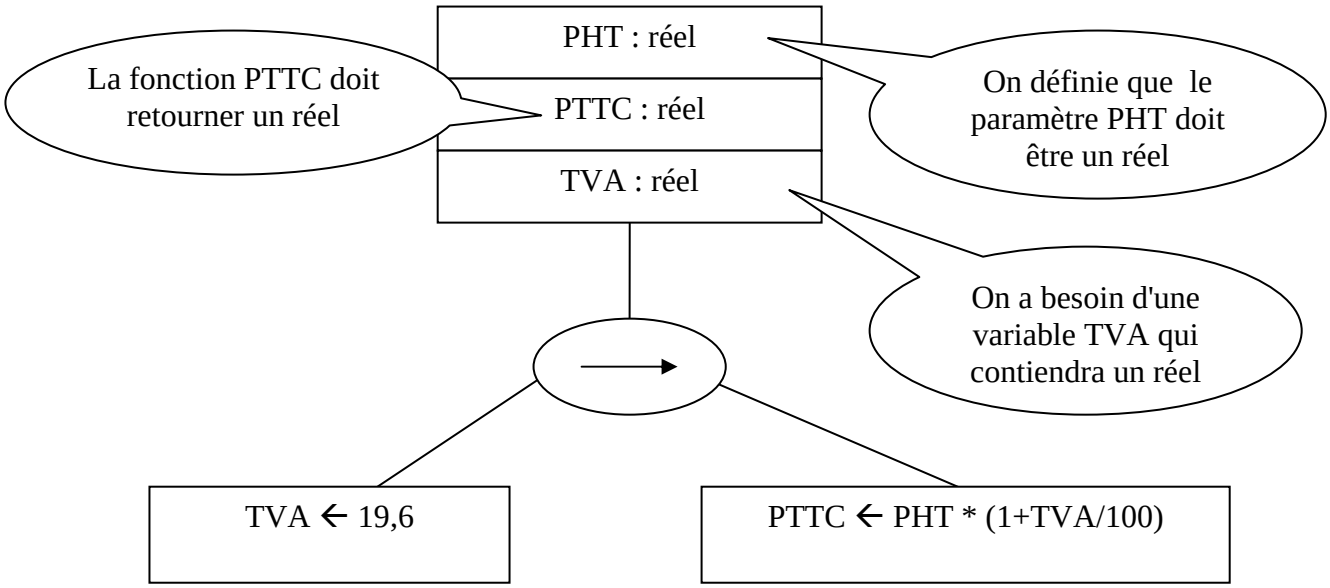
Avant d’utiliser une variable, on doit définir qu’elle genre de contenu elle pourra traiter des entiers, des réels, des caractères…(la lettre peut être une feuille, une photo, une cassette..).

5.5. Affectation (←)

A← 6 signifie que la variable de nom A reçoit le contenu 6.

5.6. Exemple

5.6.1. L'arbre



L'arbre se lit de la gauche vers la droite.

5.6.2. Le VBA

```
Function vv_pttc(pht As Single) As Single
  Dim TVA = 19,6
  vv_pttc = pht * (1 + TVA / 100)
End Function
```

5.7. Application

1) **Retourne le périmètre d'un cercle connaissant son rayon. (PI = 3,14)**
2) **Retourne la surface d'un cercle connaissant son diamètre.**
3) **Retourne une vitesse connaissant le temps et la distance.**

6. STRING

Parmi les types de variables, il existe la chaîne de caractères (string). Il est possible de combiner différentes variables afin d'obtenir un affichage différent en fonction des cas.

A ← "bonjour"	a reçoit le mot "bonjour"
b ← "tout le monde"	b reçoit "tout le monde"
c ← a & b	c reçoit "bonjour tout le monde"
c ← a & " " & b	c reçoit "bonjour tout le monde"
a ← "au revoir"	a reçoit "au revoir"
d ← a & " " & b	d reçoit "au revoir tout le monde"

le fait de combiner différentes chaînes de caractères, s'appelle la concaténation. Le & est le symbole de concaténation en VBA.

6.1. Application

1) bonjour()

Retourne le mot « bonjour ».

2) perimetre1(rayon)

Connaissant le rayon d'un cercle. Retourne « le périmètre du cercle est : ».

3) perimetre2(rayon)

Retourne « le périmètre d'un cercle de rayon R cm est de P cm ».

4) celsius(T)

Retourne en degrés Celsius une température T donnée en degrés Fahrenheit

$$celsius(T) = \frac{(T-32)\times 5}{9}$$

5) conversion_heure(NbHeures, NbMinutes, NbSecondes)

Retourne la durée en secondes.

6) moyenne_etudiant(Français, Math, Anglais)

Retourne la moyenne des notes d'un étudiant.

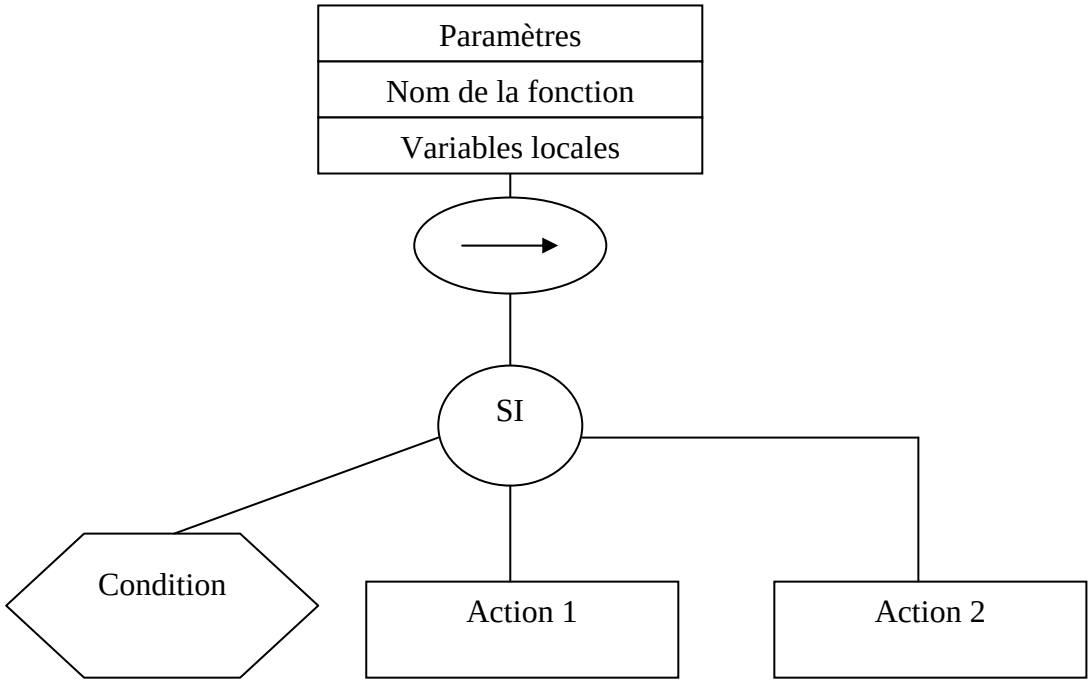
7) moyenne_etudiant2(Fran, CF,.Math, CM, Angl, CA)

Retourne la moyenne des notes d'un étudiant avec des coefficients.

7. ALTERNATIVE

Dans certain cas, le résultat retourné d'un problème résulte du choix entre deux solutions. C'est ce que l'on appelle une alternative. Ceci est matérialisé par la fonction SI.

7.1. SI simple

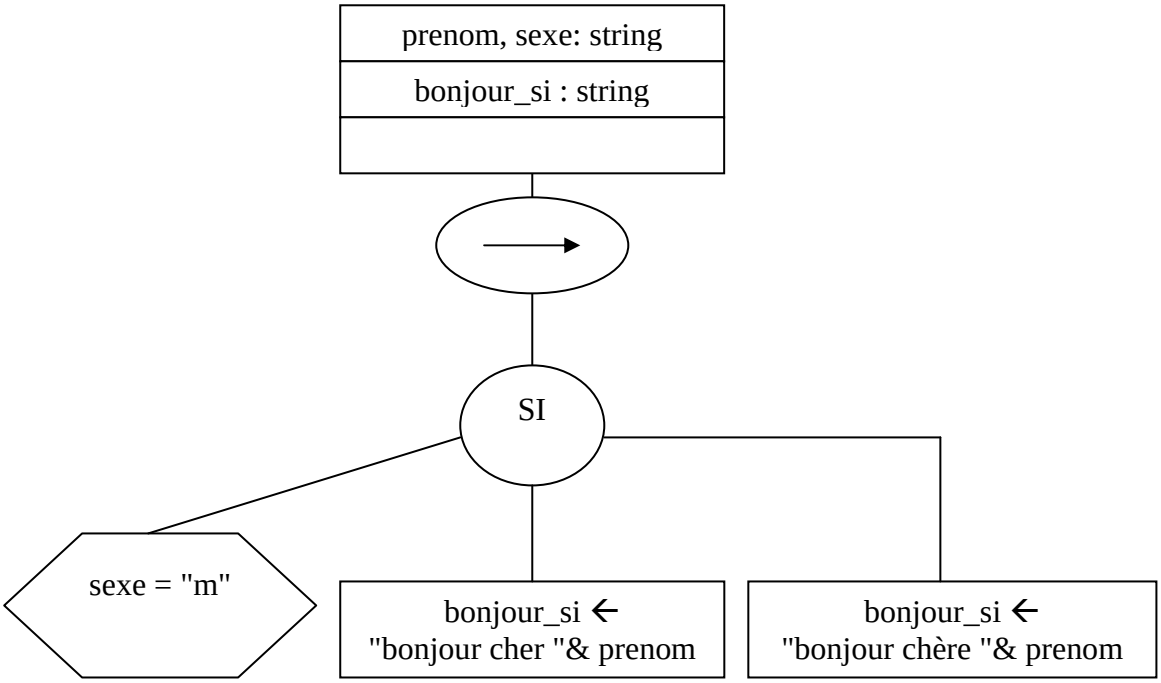


La fonction consiste à réaliser Action 1 si la condition est vraie sinon elle réalise Action 2.

7.1.1. Exemple

En fonction du prénom et du sexe d'une personne. Retourne « bonjour cher ou chère prénom »

7.1.1.1. L'arbre



7.1.1.2. Le VBA

```
Function vv_bonjour_si(prenom As String, sexe As String) As String
  If sexe = "m" Then
    Bonjour_si = "bonjour cher "& prenom
  Else
    Bonjour_si = "bonjour chère "& prenom
  End If
End Function
```

7.1.2. Application

1) code_si(code)

En fonction d'un code, si « 123 » Retourne bon code ou mauvais code
code_si(456) ➔ mauvais code

2) remise_si(montant)

Retourne un prix avec une remise de 10% si supérieur à 1000€. remise_si(2000) → 1800

3) estimation_de_prix(Prix1, Prix2)

Calcule le montant Prix1 + Prix2
Retourne « coût élevé » ou « Bon marché » suivant que le montant dépasse ou non 15€

4) absolue(nombre)

Retourne la valeur absolue d'un nombre absolue(-15) → 15

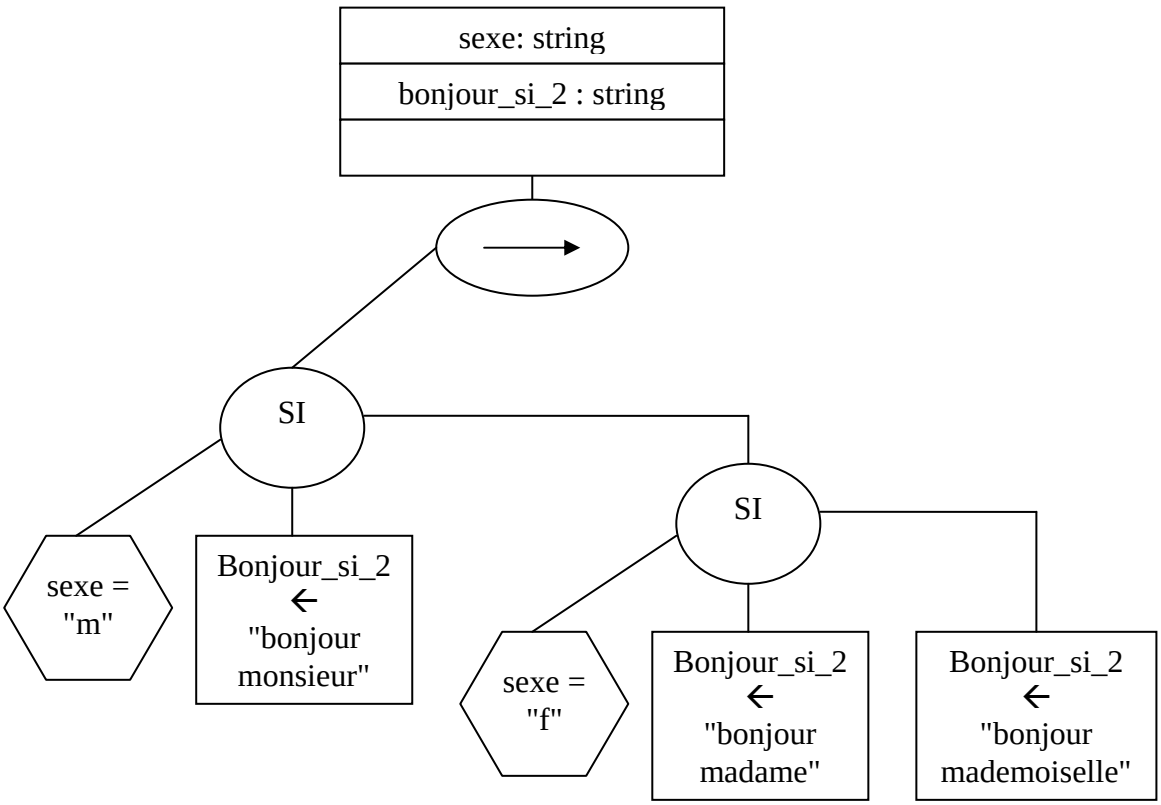
7.2. SI imbriqués

Lorsque la solution d'un problème se trouve être le choix entre plusieurs solutions, il faut soit juxtaposer les SI ou les imbriquer. Dans la première solution chaque SI sera évalué, dans le second cas dès qu'un SI sera validé les suivants ne seront pas visités.

7.2.1. Exemple

En fonction du sexe, retourne "bonjour cher monsieur"ou "madame" ou "mademoiselle"

7.2.1.1. L'arbre



7.2.1.2. Le VBA

```
Function vv_bonjour_si_2(sexe As String) As String
  If sexe = "m" Then
    Bonjour_si_2 = "bonjour monsieur"
  Else
    If sexe = "f" Then
      Bonjour_si_2 = "bonjour madame"
    Else
      Bonjour_si_2 = "bonjour mademoiselle"
    End If
  End If
End Function
```

7.2.2. Application

1) remise2_si(montant)

Retourne le prix remisé : >1000 10%, >1500 15%, >2000 20%

2) mention_diplome(moyenne)

Retourne la mention au diplôme à partir de la moyenne obtenue.
(<10) "rien" (10, 12) "Passable" (12, 14) "AB" (14 16) "B" (16, 20) "TB"

3) age(pierre, paul, jacques)

Demander l'age de pierre, Paul, jacques retourne le plus vieux.

4) equation(a, b, c)

Equation du second degré $aX^2+bX+c=0$ $delta=b^2-4ac$ $X1=(-b+\sqrt{delta})/2a$ $X2=(-b-\sqrt{delta})/2a$
equation(1,1,-2) → X1 = 1 et X2= -2

5) calcul_ttc(HT, CodeTVA)

Retourne le prix TTC du prix HT en fonction du code de TVA (1 : 5,5, 2 : 19,6, sinon retourne 0)

6) edf(redevance, MoisDeRetard)

Les pénalités appliquées en cas de paiement en retard d'une quittance EDF sont les suivantes

- Jusqu'à un mois de retard la redevance est majorée de 10%.
- De un à deux mois de retard la somme déjà redevable est majorée de 12.5%
- de deux a trois mois de retard la somme déjà redevable est majorée de 15% »

edf(100 ; 1) = 110
edf (100 ; 2) = 123,75
edf (100 ; 3) = 142,31

7) cout_de_transport(Poids, Volume, MoyenDeTransport>

Lorsqu'un transporteur achemine de la marchandise, le taux de base du prix de transport dépend principalement du moyen de transport utilisé :

- Le fret aérien est facturé au poids;
- Le fret maritime est facturé au volume ;
- Le fret ferroviaire est facturé à la fois au poids et au volume.

Par air : 0,08 € le kg, Par mer : 100 € Le m3, Par chemin de fer : 5 € le m^3, plus 0,03 € par kg

cout_de_transport (2900 ; 2,75 ; « Air ») = 232

8) banque(Montant, NbAnnée)

Une banque propose un mode de placement à taux d'Intérêt progressif sur trois ans.

- 3,6% la première année
- 5,4% la seconde année
- 7,2% la troisième année

Montant représente le montant initial du placement et NbAnnée le nombre d'années d'épargne.
banque(1500 ; 1) = 1554 banque(1500 ; 2) = 1637,92 banque(1500 ; 3) = 1755,85

8. BOOLEEN

Un booléen est une variable logique qui ne peut prendre que la valeur **vraie** ou **fausse**.

8.1. Expression logique

Une expression logique est la combinaison d'affirmations vraies ou fausses qui renvoie une réponse vraie ou fausse. Les relations d'égalité et d'inégalité (=, <, >) ainsi que les opérateurs logiques (ou, et, non) permettent de créer l'ensemble de ces expressions.

8.2. Symboles de comparaisons

=	égalité
<	inférieur
>	supérieur
<=	inférieur ou égal
>=	supérieur ou égal
<>	différent

8.3. Opérateurs logiques

OU (OR)

VRAI	**OU**	VRAI	→	VRAI
VRAI	**OU**	FAUX	→	VRAI
FAUX	**OU**	VRAI	→	VRAI
FAUX	**OU**	FAUX	→	FAUX

ET (AND)

VRAI	**ET**	VRAI	→	VRAI
VRAI	**ET**	FAUX	→	FAUX
FAUX	**ET**	VRAI	→	FAUX
FAUX	**ET**	FAUX	→	FAUX

NON (NOT)

NON	VRAI	→	FAUX
NON	FAUX	→	VRAI

OU exclusif (XOR)

VRAI	**XOR**	VRAI	→	FAUX
VRAI	**XOR**	FAUX	→	VRAI
FAUX	**XOR**	VRAI	→	VRAI
FAUX	**XOR**	FAUX	→	FAUX

8.4. Priorités

Une opération mathématique s'exécute suivant la priorité des opérateurs. L'opération 4+2*3 correspond à l'addition de 4 au produit de 2 et 3. Les différents opérateurs ont aussi un ordre de priorité.

1	Le signe moins	-
2	La négation	NOT
3	multiplication et division	*, /, AND
4	addition et soustraction	+, -, OR, XOR
5	comparaisons	=, <, >, <=, >=, <>

De plus les expressions placées entre parenthèses sont évaluées en premier. Si des parenthèses sont imbriquées, les expressions les plus internes sont prises en compte les premières.

8.5. Exemples

a ← 2 b ← 3 c ← 4

(a > 3) .... faux
(a<4) et (b<4) ..... vrai et vrai .... vrai
(non (a> 3)) ou (c<2)..... (non faux) ou faux..... vrai ou faux ..... vrai

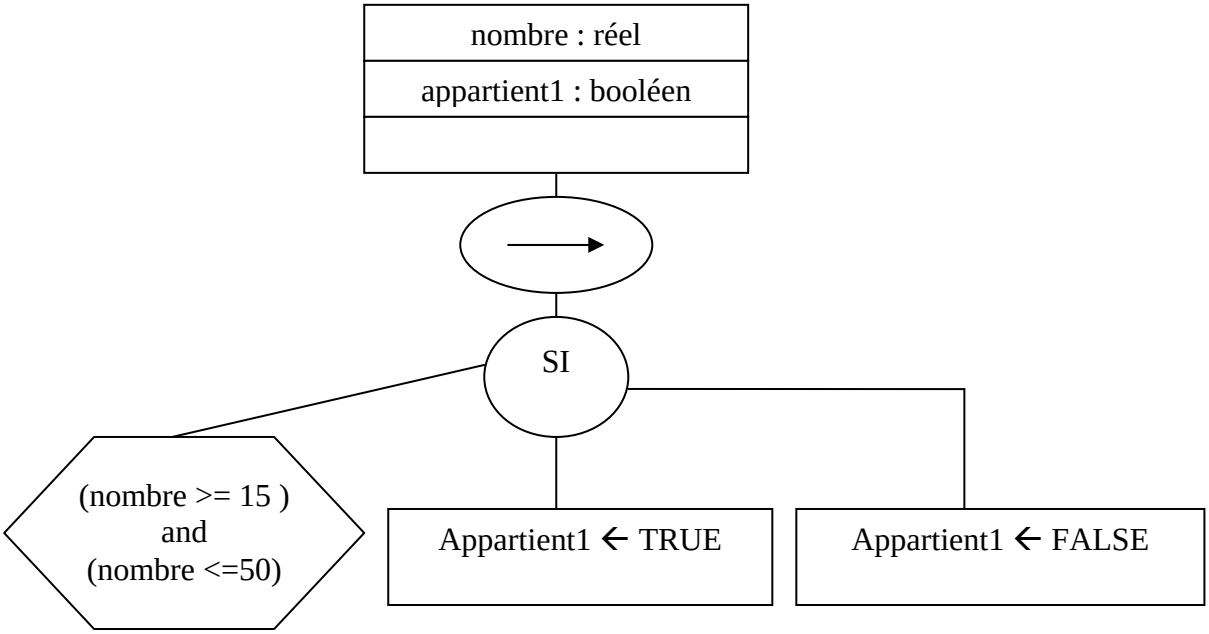
```
(3+2)*4+2      <=     (5* 4+2)-5
   5*4+2       <=     (20+2)-5
    20+2       <=     22-5
      22       <=     17
               FAUX
```

```
NOT   (3<2)   AND   (2=2)   OR   NOT   (5>4)
NOT   FAUX    AND   VRAI    OR   NOT   VRAI
      VRAI    AND   VRAI    OR   FAUX
                    VRAI    OR   FAUX
                            VRAI
```

8.6. Exemple

Retourner si un nombre appartient à [15 ;50]

8.6.1. L'arbre



8.6.2. Le VBA

```
Function vv_appartient1(nombre As Single) As Boolean
  If (nombre > 15) AND (nombre < 50) Then
    Appartient1 = TRUE
  Else
    Appartient2 = FALSE
  End If
End Function
```

8.7. Application

1) Appartient2(nombre)

Retourne si un nombre n'appartient pas à [10 ;100]

2) Appartient3(nombre)

Retourne si un nombre appartient à [15 ;50].∪ [100 ; +∞[

3) Age2(pierre,paul,jacques)

En fonction de l'age de Pierre, Paul et Jacques. Retourne le plus vieux et le plus jeune

4) Impaire_5_7(nombre)

Retourne si un nombre est impair et divisible par 5 et pas par 7 (voir MOD).

5) Bissextile(annee)

Retourne si l'année est bissextile. (Divisible par 4 sauf les années séculaires pas divisibles par 400. 1984 et 1600 oui 1700 non.)

9. SOUS ROUTINE

9.1. MsgBox

Voir l'aide d'Excel

9.2. InputBox

Voir l'aide d'Excel

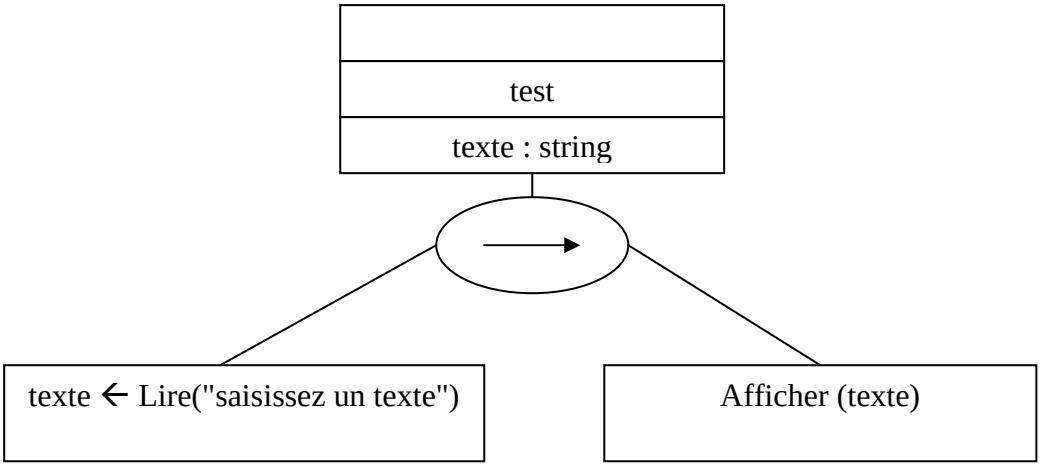
9.3. Sub

Il existe deux sortes de procédure, les fonctions et les sous-routines. Les fonctions permettent de trouver des solutions et de renvoyer une valeur. Quand aux sous-routines, elles réalisent des taches mais ne renvoient pas de valeur.

9.3.1. Exemple

Ouvre une boite de saisie demandant d'écrire un message, puis ouvre une boite de dialogue contenant ce message.

9.3.1.1. L'arbre



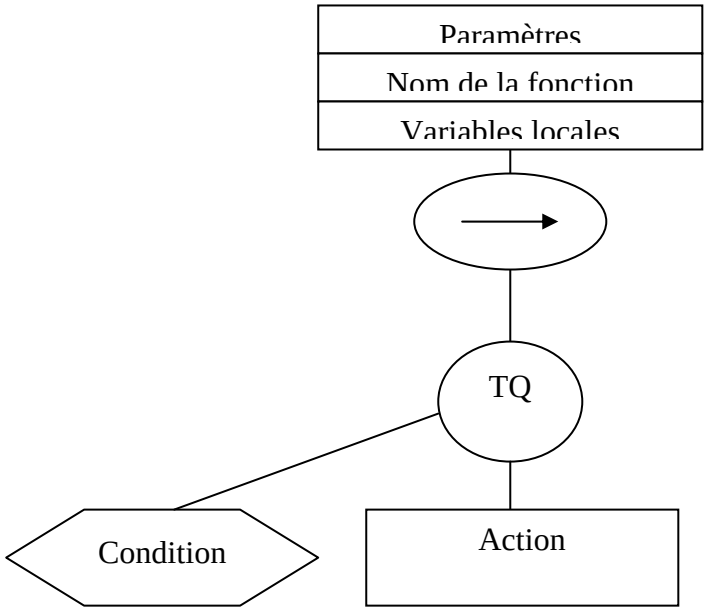
9.3.1.2. Le VBA

```
Sub vv_test()
  Dim texte As String
  texte = InputBox("coucou")
  MsgBox (texte)
End Sub
```

10. ITERATIVE

Pour certain problème, on a besoin de répéter l'action plusieurs fois. Soit tant qu'une condition est vraie soit pour un nombre de fois bien précis. Cette action s'appelle une itération.

10.1. Tant Que

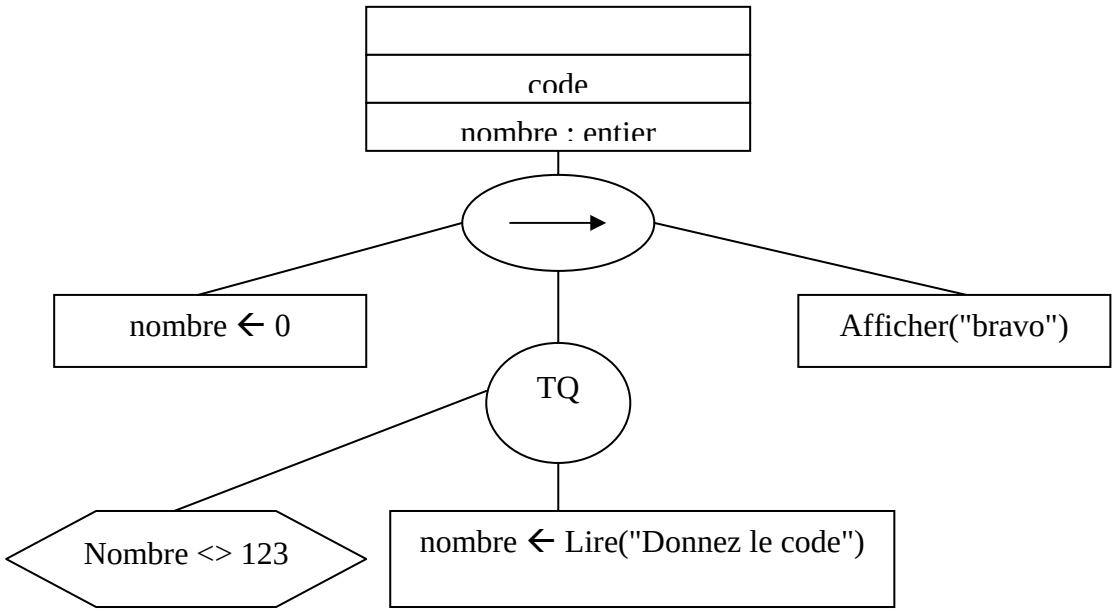


La fonction réalise l'action tant que la condition est vraie

10.1.1. Exemple

Demander un code jusqu'à ce qu'il soit trouvé

10.1.1.1. L'arbre



10.1.1.2. Le VBA

```
Sub vv_code()
  Dim nombre As Integer
  Nombre = 0
  While nombre <> 123
    nombre = InputBox("Donnez le code")
  Wend
  MsgBox ("bravo")
End Sub
```

10.1.2. Application

1) Recherche_1()

Retourne "bravo" si on retrouve un nombre donné par un premier utilisateur

2) Recherche_2()

Idem plus donne le nombre de coup effectué.

3) Recherche_3()

Idem plus 10 propositions max.

4) Somme_nombre()

Demande des nombres. Arrête quand chiffre 0. Affiche la somme.

5) Moyenne_nombre()

Demande des nombres. Arrête quand chiffre 0. Affiche la moyenne

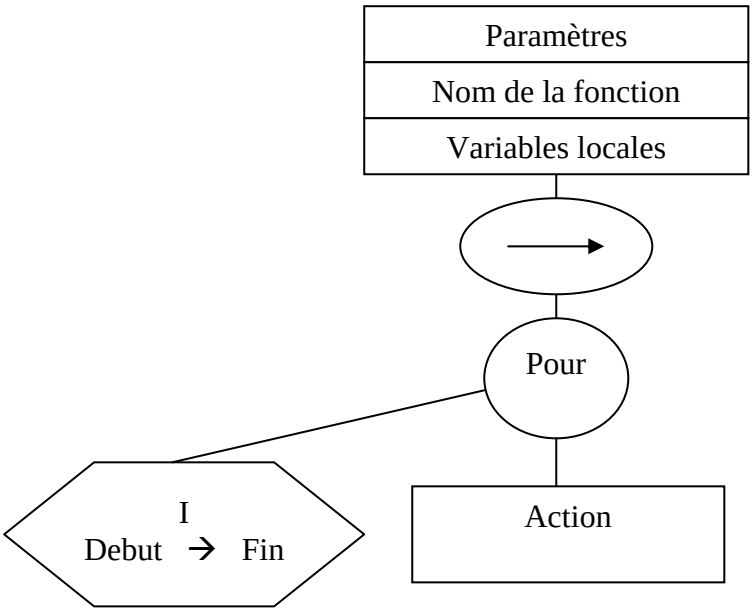
6) Max_Min()

Demande des nombres. Arrête quand chiffre 0. Affiche le max, le min.

7) Demande_tq()

Demande des nombres tant qu'ils sont différents de 0 ou tant que la somme est plus petite que 100.

10.2. Pour

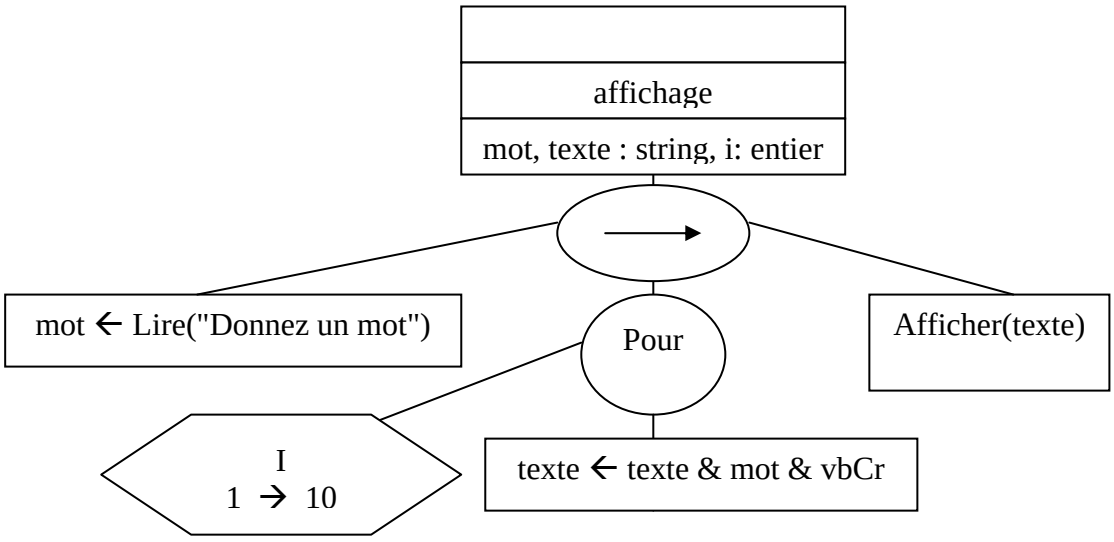


La fonction réalise l'action pour I variant de début à fin. I est un entier qui s'incrémente de 1 à chaque fois que l'on exécute l'action.

10.2.1. Exemple

Demander un mot l'écrire 10 fois.

10.2.1.1. L'arbre



10.2.1.2. Le VBA

```
Sub vv_affichage()
  Dim mot As String
  Dim texte As String
  Dim i As Integer
  mot = InputBox("Donnez un mot")
  For i = 1 To 10
    texte = texte & mot & vbCr
  Next i
  MsgBox (texte)
End Sub
```

10.2.2. Application

1) Affiche_1_20()

Affiche les nombres de 1 à 20.

2) Affiche_5_9()

Affiche les nombres de 5 à 9.

3) Table_8()

Affiche la table de 8

4) Table_n(n)

Affiche la table de multiplication du paramètre n.

5) Produit_ab(a, b)

Retourne le produit de a et b, avec seulement l'addition.

6) Puissance_ab(a, b)

Retourne a à la puissance b avec seulement la multiplication.

7) Factoriel(n)

Retourne le factoriel de n.

8) Affiche_paire()

Affiche les nombres pairs de 1 à 100.

9) Affiche_3()

Affiche les nombres divisibles par 3 entre 1 et 100.

10) Max_Min_10()

Demande 10 chiffres. Affiche le max et le min.

11) Affiche_10_lignes()

Affiche les nombres de 1 à 10, 10 fois, sur 10 lignes.

12) Affiche_100_lignes()

Affiche les nombres de 1 à 100 sur 10 lignes.

13) Affiche_10()

Affiche les chiffres de 1 à 10, puis de 1 à 9, puis de 1 à 8, jusque 1.

11. TABLEAUX

Les tableaux permettent de regrouper des données dans la même variable. (Coordonnées de vecteurs, listes de nombres, listes de caractères, listes de mots…).

11.1. Tableaux unidimensionnels

Tableau d'entier : TABI

125	34	8	12	765	23	42	1

Pour accéder à une case du tableau il suffit de la désigner par son emplacement :
TABI(5) {correspond à la cinquième case du tableau TABI c'est à dire 765 }
TABI(8) ← 1 {met le chiffre 1 dans la huitième case du tableau TABI}

i ← 4 ;
TABI(i) ← 12 {met le chiffre 12 à la quatrième place du tableau TABI }

11.2. Tableaux de tableaux

On peut déclarer des tableaux dont chaque élément est lui-même un tableau. Ce qui revient à déclarer un tableau à plusieurs dimensions

Exemple :

tab peut être considéré comme une matrice de 3 colonnes et 2 lignes

tab2(1,1)	tab2(2,1)	tab2(3,1)
tab2(1,2)	tab2(2,2)	tab2(3,2)

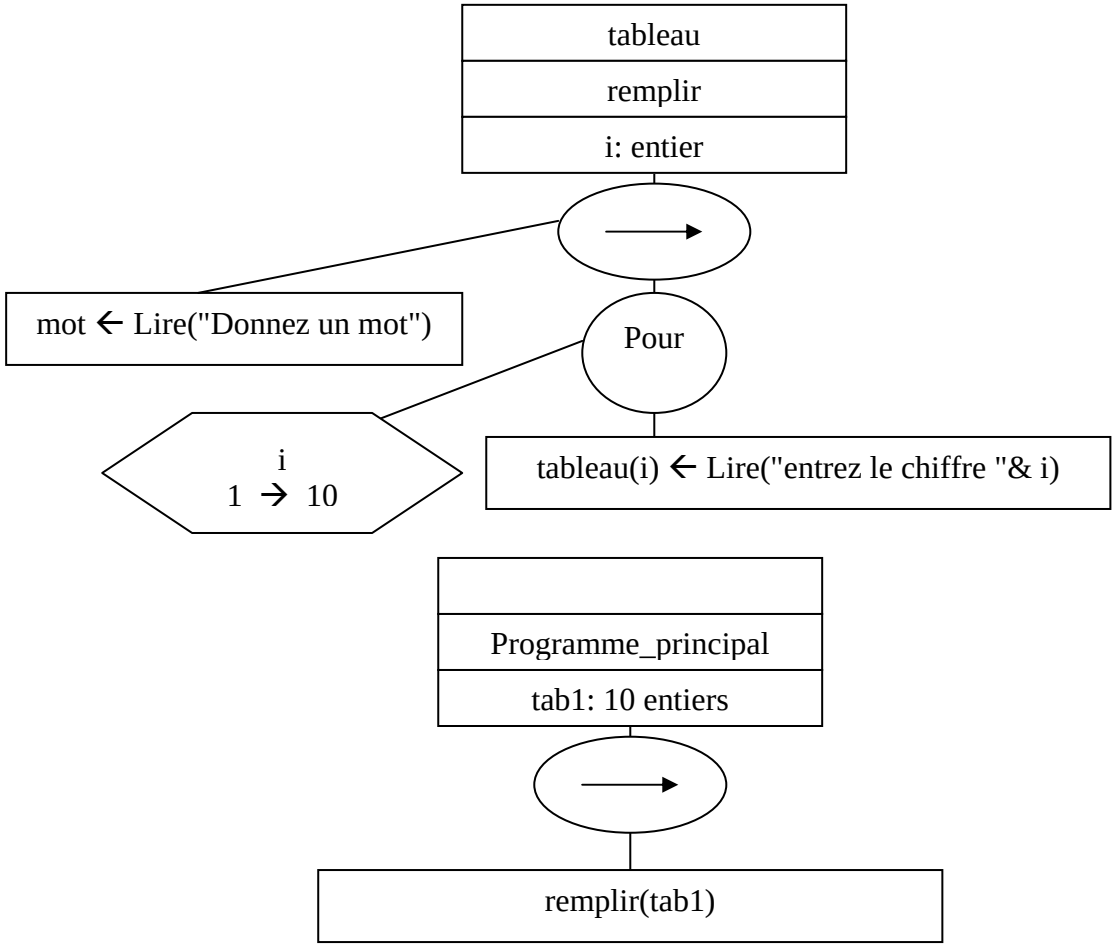
11.3. En VBA

```
Dim NomVariable(taille|Début to Fin,taille2|Début2 to Fin2,…)
```

11.4. Exemple

Remplir un tableau de 10 nombres. Appeler cette sous routine à partir d'une autre

11.4.1. L'arbre



11.4.2. Le VBA

```
Sub vv_remplir(tableau)
  Dim i As Integer
  Dim texte As String
  For i = 1 To 10
    tableau(i) = InputBox("entrez le chiffre " & i)
  Next i
  For i = 1 To 10
    texte = texte & tableau(i) & vbCr
  Next i
  MsgBox (texte)
End Sub

Sub programme_principal()
  Dim tab1(1 To 10) As Integer
  Call vv_remplir(tab1)
End Sub
```

11.5. Application

1) Affiche_tab(tableau)

Affiche le contenu de tableau.

2) Moyenne_tab(tableau)

Retourne la moyenne des nombres d'un tableau.

3) Appartient_tab(tableau)

Demande un nombre et retourne vrai ou faux si celui-ci appartient au tableau.

4) Place_tab(tableau, nombre)

Affiche la ou les places de nombre dans le tableau.

5) Echange_tab(tableau, rang1, rang2)

Echange le contenu de deux cases du tableau.

6) Max_tab(tableau)

Retourne la place du max.

7) Impaire_tab(tableau)

Retourne le nombre de chiffre impair dans le tableau.

8) Somme_3(tableau)

Retourne la somme des chiffres divisibles par 3 du tableau.

9) Affiche_tab_0(tableau)

Affiche le contenu du tableau jusqu'au nombre 0.

10) Remplace_double(tableau)

Remplace chaque nombre d'un tableau par son double

11) Remplace_01(tableau)

Remplace chaque nombre du tableau par 0 s'il est pair, par 1 s'il est impair

12) Remplir_tab2(tab1, tab2)

Remplie tab1 avec le carré des nombres de tab2

13) Somme_p_I(tableau)

Affiche la somme des nombres pairs et la somme des nombres impairs du tableau.

14) Affiche_tab_p_i(tableau)

Affiche les nombres impairs sur une ligne et les nombres pairs sur une autre.

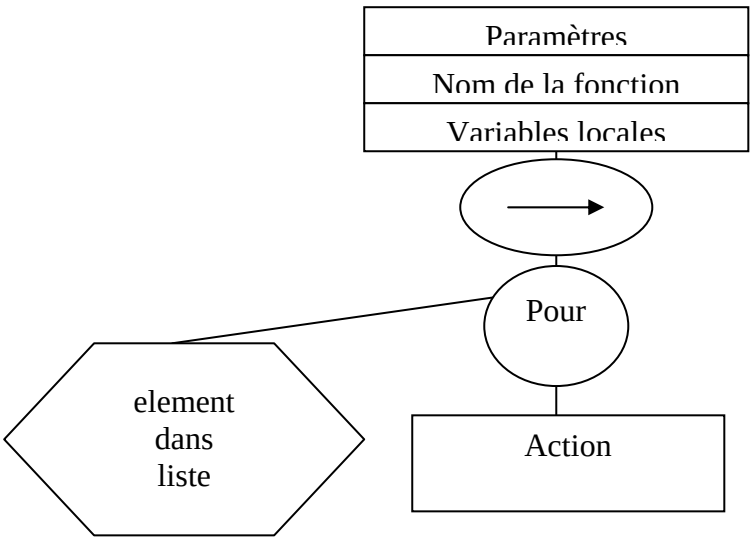
15) Affiche_en_lettres(tableau)

A la place des chiffres 1, 2, 3, 4 affiche un, deux, trois, quatre.

16) Affiche_puissance(tableau)

Affiche sur la première ligne, le premier chiffre à la puissance 1, sur la deuxième le deuxième à la puissance 2 etc.

12. POUR ELEMENT DANS LISTE

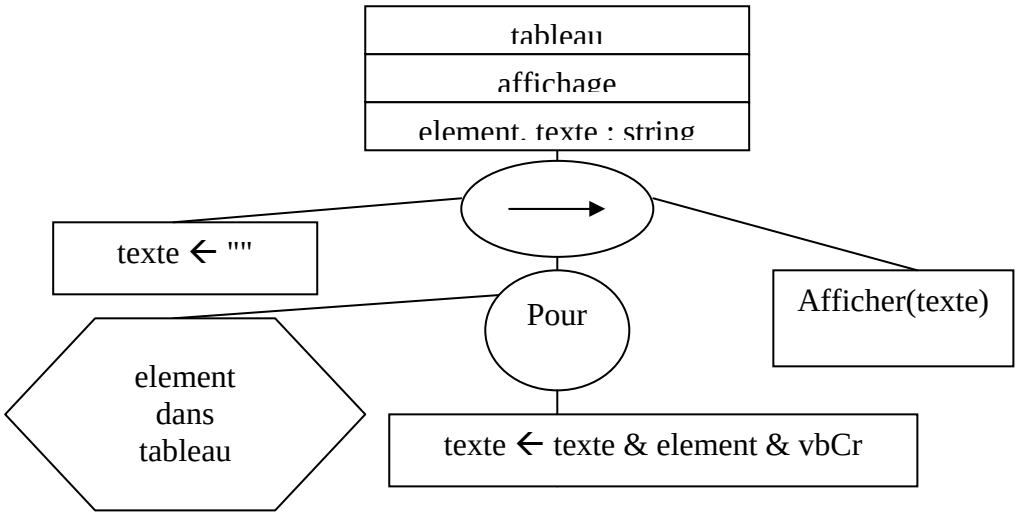


La fonction réalise l'action autant de fois qu'il y a d'objet dans la liste. A chaque tour la variable element prend le contenu de l'objet suivant de la liste.

12.1. Exemple

Afficher le contenu d'un tableau.

12.1.1. L'arbre



12.1.2. Le VBA

```
Sub vv_affichage(tableau)
  Dim texte As String
  Dim element As String
  For Each element In tableau
    texte = texte & element & vbCr
  Next element
  MsgBox (texte)
End Sub
```

12.2. Application

Reprendre les exercices précédents.

13. CHAINE DE CARACTERES

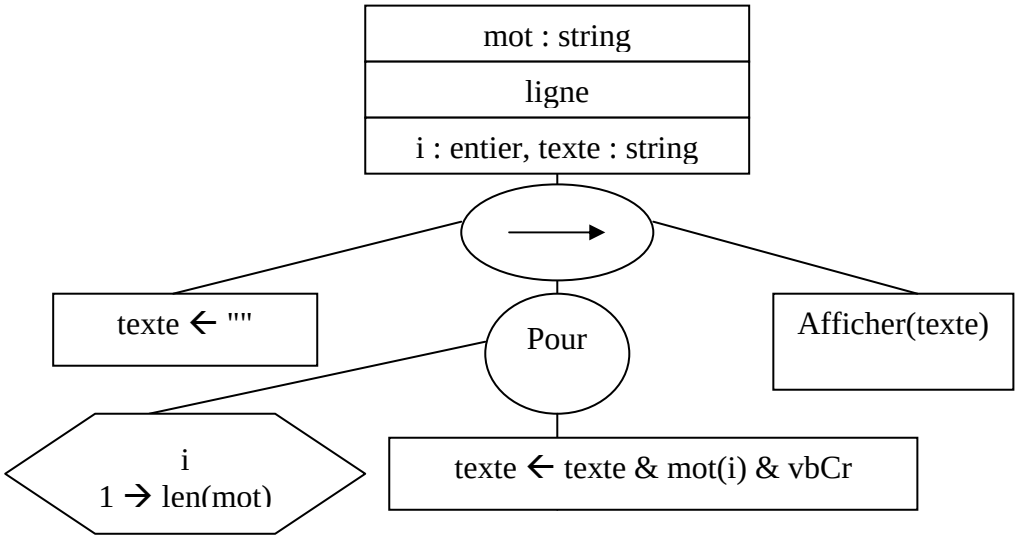
13.1. Mid, Len

Voir l'aide

13.2. Exemple

Afficher une lettre d'un mot par ligne.

13.2.1. L'arbre



13.2.2. Le VBA

```
Sub vv_ligne(mot as string)
  Dim i As Integer
  Dim texte As String
  texte = ""
  For i = 1 To Len(mot)
    texte = texte & Mid(mot, i, 1) & vbCr
  Next i
  MsgBox (texte)
End Sub
```

13.3. Application

1) N_fois(mot)

Affiche autant de fois le mot qu'il a de lettres.

2) N_premiere(mot, n)

Affiche les n premières lettres du mot.

3) Envers(mot)

Affiche le mot à l'envers.

4) Decalage(mot)

Affiche une lettre par ligne décalée d'un espace par rapport à la précédente.

5) Palindrome(phrase)

Retourne si c'est un palindrome.

6) Places(mot, lettre)

Retrouve et affiche, la ou les places de la lettre dans le mot.

7) Occurrence(mot, lettre)

Retourne le nombre de fois que la lettre apparaît dans le mot.

8) Voyelle(mot)

Retourne le nombre de voyelles dans le mot.

9) Remplace_voyelle(mot)

Retourne le mot avec ses voyelles remplacées par la voyelle i.

10) Supprime_lettre(mot)

Retourne le mot sans la deuxième lettre.

11) Supprime_voyelle(mot)

Retourne le mot sans ses voyelles.