

# CORRESPONDANCE

## LANGAGE NF04 – LANGAGE VB

### Déclarations

	NF04	VB
Commentaire	// <commentaire sur la ligne>	Rem ou ' (guillemet simple)
Déclaration des variables	<variable> : <type>  Sensible à la casse	option explicit Dim <variable> As <type>  Insensible à la casse
Déclaration des Tableau de réels/ d'entiers	<nom_du_tableau> : tableau [<constante (entier)>... <constante (entier) >] de <type> Rq : indice de début =0/1	Dim <variable> (<constante (entier)>To <constante (entier) >] As <type>  Rq : indice de début =0 ou 1
Chaîne de caractères	C'est un tableau de caractères	C'est un tout Dim <variable> As String
Programme	Algorithme <nom> Variables <declaration variables> Instructions <liste d'instructions> Fin <nom>	Sub <nomprog> () ' Variables <declaration variables> ' Instructions <liste d'instructions> End Sub

### Types de variables

	NF04	VB
Booléen	booléen	boolean
Entiers (2 octets) 32 768 à 32 767 (4 octets) -2 E9 +2 E9	entier	Integer (2 octets) Long(4 octets)
Réels (4 octets) E-45 à E38 (8 octets) E-324 à E308	réel	Single Double
Caractère	caractère	

### Opérateurs

	NF04	VB
Addition, soustraction, multiplication, division	+, -, *, /	+, -, *, /
Division entière	E(/)	
modulo	%	mod
Egalité	=	=
infériorité stricte, infériorité, supériorité stricte, supériorité	<, ≤, >, ≥	<, <=, >, >=
différence	≠	<>
OU logique	OU	OR
ET logique	ET	AND

## Instructions

NF04	VB
Affectation <variable> ← <expression>	<variable> =<expression >
Si <expression (booléen)> Alors <instruction 1> <instruction N> FinSi	If <expression (booléen)> Then <instruction 1> <instruction N> End If
Si <expression 1> Alors <liste d'instructions 1> SinonSi <expression N> Alors <liste d'instructions N> Sinon <liste d'instructions N+1> FinSi	If <expression 1 > Then <liste d'instructions 1> Elseif <expressionN > Then <liste d'instructions N> Else <liste d'instructions N+1> End If
Pour <variable (entier)> de <expression (entier)> à <expression (entier)> par pas de <expression (entier)> <liste d'instructions> FinPour	For < variable > = <expression (entier)> To <expression (entier)> [Step <expression (entier)> ] <liste d'instructions > Next
Tant que <expression (booléen)> faire <liste d'instructions > Fintq	Do While <expression> <liste d'instructions> Loop
Lire (clavier ! <variable>)	<variable> = InputBox(<message>)
Écrire ("<message à afficher>",<expression> !)	Call MsgBox("<message à afficher>" & <expression>)
Valeur d'une cellule dans le tableur (en lecture ou écriture)	Range("C1").Value Ou Cells(1, 3).Value

### Remarques :

- Pas de distinction entre les variables n et N (Insensible à la casse)
- L'opérateur & permet la concaténation de chaînes de caractères
- Effacer des cellules : Range("A1","A100").Clearcontents