

# Annexe 01 : Traduction des instructions algorithmiques en VB

## Introduction

Le but de cette annexe est de traduire les instructions algorithmiques vues dans le chapitres 1 jusqu'au chapitre 7 en VB.

## Traduction des instructions du chapitre 01

Instruction algorithmique			Instruction Visual Basic		
Variable i : entier			Dim i as integer		
Constante PI = 3.14			Const PI = 3.14		
Réel	Chaîne	Logique	Double	String	Boolean
Caractère	Chaîne*20		String* 1	String*20	
Ecrire("message")			Msgbox "message"		
NB :					
- La virgule « , » dans <i>Ecrire</i> devient & pour la fonction <i>Msgbox</i> ;					
- Voir l'annexe 2 pour plus d'information sur MsgBox et InputBox.					
Ecrire("La valeur de x est ", x)			Msgbox "La valeur de x est " & x		
Lire(i)			i = inputbox("")		
Ecrire("Donnez i ")			i = inputbox("Donnez i ")		
Lire(i)					
$i \leftarrow 4$			i = 4		
Div	Mod	Et	\	Mod	And
Ou	Ou <sub>ex</sub>	exposant	Or	Xor	^
Non	≠	=	Not	◊	=
Ecrire("\n")			VBCRLF Ou bien chr(13) + chr(10)		
Ecrire("Le résultat est ", R)			Msgbox "Le résultat est " & R		
Ecrire(A, B, C)			Msgbox A & B & C		

Lire(A, B, C)		A = inputbox() B = inputbox() C = inputbox()	
Vrai	Faux	True	False
// Commentaire		' Commentaire	

**Note :**

En Visual Basic, il n y a pas de différence entre majuscule et minuscule.

**Traduction des instructions du chapitre 02**

Instruction algorithmique	Instruction Visual Basic		
Si ... alors ... Sinon ... FinSI	If ... then ... Else ... End if		
Si ... alors ... SinonSi ... alors ... Sinon ... FinSI	If ... then ... Elseif ... then ... Else ... End if		
Selon ... faire ... Sinon ... FinSelon	Select case ... ... Case Else ... End Select		
Pour i ← 1 jusqu'à 10 ... FinPour	For i = 1 to 10 ... Next i		
Pas	Quitter Pour	Step	Exit For

TantQue ... Faire ... FinTantQue	Do While ... ... Loop
Répéter ... Jusqu'à ...	Do ... Loop Until ...

### Traduction des instructions du chapitre 03

Instruction algorithmique	Instruction Visual Basic
Variable T : tableau[1 .. 5] de réels	Dim T(1 to 5) as double
Variable T : tableau[1..5, 1..10] de réels	Dim T(1 to 5, 1 to 10) as double

### Traduction des instructions du chapitre 04

Instruction algorithmique	Instruction Visual Basic
Type structure personne Nom : chaîne Prenom : chaîne EndStruct	Type personne Nom : string Prenom : string End Type
Variable e : personne	Dim e as personne
e.Nom ← "Sabir"	e.Nom = "Sabir"

### Traduction des instructions du chapitre 05

Instruction algorithmique	Instruction Visual Basic
Fonctions liées aux chaînes de caractères	
Longueur(ch1)	Len(ch1)
Concat(ch1, ch2)	ch1 & ch2 ou bien ch1 + ch2
Copie(ch1, position, n)	Mid(ch1, position, n)

Recherche(ch1,ch2)			Instr(ch1,ch2)		
Comp(ch1,ch2)			StrComp(ch1,ch2)		
Fonctions mathématiques et trigonométriques					
Sqrt(x)	Abs(x)	Ent(x)	Sqr(x)	Abs(x)	Int(x)
Cos(angle)	Sin(angle)	Tan(angle)	Cos(angle)	Sin(angle)	Tan(angle)
Alea()			Rnd()		

## Traduction des instructions du chapitre 06

<b>Déclaration d'une procédure</b>	
<b>Instruction algorithmique</b>	<pre>// Déclaration d'une procédure Procédure nom_proc(p1: type, p2 : type, ...) Variables x1 : type ... ,xn : type Début     Instruction(s) FinProc</pre>
<b>Instruction Visual Basic</b>	<pre>' Déclaration d'une procédure Sub nom_proc(p1 as type, p2 as type, ...)     Dim x1 as type, ... ,xn as type     Instruction(s) End sub</pre>

<b>Déclaration d'une fonction</b>	
<b>Instruction algorithmique</b>	<pre>// Déclaration d'une fonction Fonction nom_Fonct(p1: type, p2 : type, ...) : type Variables x1 : type ... ,xn : type Début     Instruction(s)     Retourner expression FinFonct</pre>

<b>Instruction Visual Basic</b>	' Déclaration d'une fonction Function nom_Fonct(p1 as type, p2 as type, ...) as type Dim x1 as type, ... , xn as type Instruction(s) nom_Fonct = expression End Function
---------------------------------	---

<b>Appel d'une procédure</b>	
<b>Instruction algorithmique</b>	Nom_proc(liste_de_paramètres)
<b>Instruction Visual Basic</b>	<i>Call</i> Nom_proc (liste_de_paramètres) Ou bien Nom_proc liste_de_paramètres

<b>Appel d'une fonction</b>	
<b>Instruction algorithmique</b>	Nom_Fonct(liste_de_paramètres)
<b>Instruction Visual Basic</b>	<i>Call</i> Nom_fonct (liste_de_paramètres) Ou bien Nom_fonct liste_de_paramètres Ou bien y = Nom_fonct (liste_de_paramètres)

## Traduction des instructions du chapitre 07

<b>Traitement séquentiel des fichiers texte</b>			
<b>Instruction algorithmique</b>		<b>Instruction Visual Basic</b>	
Ouvrir nom_du_Fichier en Num_canal en Lecture		Open nom_du_Fichier For Input As # Num_canal	
Ecriture	Ajout	Output	Append

<pre>// Lecture d'un fichier LireFichier Num_canal, Var1</pre>	<pre>'Lecture du fichier ligne par ligne Line Input # Num_canal, Var1 'Lecture du fichier en une seule fois dans la variable Var1 Var1 = Input(LOF(Num_canal), Num canal)</pre>
<pre>EcrireFichier Num canal, Var1</pre>	<pre>Print # Num canal, Var1</pre>
<pre>Fermer (Num canal)</pre>	<pre>Close # Num canal</pre>

**Remarque :**

Le chapitre 8 ne contient pas de nouvelles instructions algorithmiques à traduire.

**Conseil :**

On vous conseille de consulter l'ouvrage intitulé :

**« Apprendre à programmer : Algorithmique & programmation VB »  
Cours, Ateliers Pratiques & Examens Corrigés (CD-ROM inclus)**

Auteurs : O. El Kharki, J. Mechbough & D. Ducrot.

Ce livre traduit et implémente les exercices de chaque atelier en VB. Le CD-ROM de ce livre contient le **code source** et l'**exécutable** de chaque algorithme.

Veuillez nous contacter par email ([mechbough@hotmail.com](mailto:mechbough@hotmail.com)) si vous ne trouvez pas nos livres dans votre ville.