

The background features a large, light gray logo for 'ISTP' (Institut Supérieur des Techniciens Professionnels) and the word 'INGÉNIEURS' at the bottom. To the right, there is a stylized, light blue figure of a person with arms raised in a 'V' shape.

ALGORITHMIE

Partie VBA

Démo application VBA

Evaluation sur la base d'un DM réalisé à partir d'un cahier des charges :

- Travail Effectué en groupe de 4 personnes.
- Remise d'un dossier de pseudo code.
- Remise d'une application en VBA.

Cours 1: Méthode de réalisation d'un algorithme. Introduction au pseudo code.

Cours 2: mise en œuvre d'algorithme en pseudo code avec AlgoBox. Découverte de l'environnement de développement VBA.

Cours 3: Ecriture d'algorithmes en VBA

Cours 4: Découverte des formulaires en VBA.

Cours 5: Réalisation d'une application avec des formulaires en VBA.



LES VARIABLES

INGÉNIEURS

• Descriptions

Taille: optimisée pour économiser ressources PC

numérique

- byte (0 à 255)
- entier simple (-32 768 à 32767)
- entier long (-2147483648 à 2147483647)
- réel simple ($-3,4 \times 10^{38}$ à $3,4 \times 10^{38}$)
- réel double (-1.79×10^{308} à 1.79×10^{308})

monétaire

- 2 chiffres après la virgule, devise

date

- numéro de série
- format

alphanumérique (entre guillemets: " ")

- texte
 - caractères
 - chaîne
- nombre sous forme de texte (code postal)

booléen

- vrai/faux
- oui/non
- 0/1

exemples

- Var A :byte
- Var coefA: réel double
- Var prixHT: monétaire
- Var dateEffet: date (jj/mm/yyyy)
- Var indiceBruit: caractères
- Var nomFamille: string,
calculAutorise: bool

• Descriptions

Portée-Durée

accessibilité

- publique: Projet
- privée: Zone de déclaration

conservation

- constante: initialisée 1 fois
- static: conserve la valeur précédente

Exemples

- publique si déclarée avant la procédure
- Const Entier nbMachines \leftarrow 10
- Static Entier tentativesMDP \leftarrow 3

• Descriptions

Spéciales

tableaux

- spécialisés
- variant

objets

- sélections
- plage
- feuille Excel, classeur, ...

énumératives

- regroupement de plusieurs constantes ordonnées

structurées

- regroupement de types différents

intervalles

- définition de valeurs ordonnées et bornées au mini et maxi

Exemples

- Tableau Entier: t(10) \\10 cases d'entiers
- Tableau String: tab(3) \\3 cases de textes
- Tableau Entier tab(5,6) \\(5 lignes, 6 colonnes) d'entiers
- Tableau tab \\variant, taille non définie

- monClasseur: objet classeur Excel

- Enum droits

lecture=-1

ecriture=0

lecture_ecriture=1

FinEnum \\ lecture < ecriture < lecture_ecriture

- Struct Famille

Var Nom: string

Var Code: string

Var nbSousEnsembles: byte

Var nbArticles: entier

Var statutActif: bool

FinStruct

- JourOuvrés: Lundi..Vendredi

Pourcentage: 0..100

IndiceBruit: A..E \\descriptif, inconnu en vba hormis range(selection), ou tableau



FONCTIONS

INGÉNIEURS

• Principales fonctions textes

taille(chaine): nb caractères

gauche(chaine, n): n car depuis gauche

droite(chaine, n): idem depuis droite

extraire(chaine, depart, n): extrait une partie de la chaine commençant au caractère de départ et long de n caractères

remplacer(chaine,texte1,texte2): remplace dans une chaine la chaine trouvée texte1 par la texte2

trouve(chaine1,chaine2): renvoie la position du caractère qui commence la chaine dans la chaine1 ou erreur si pas trouvé

Equivalent VBA

- len(string)
- left(string, n)
- right(string,n)
- mid(string,n1,n2)
- replace(string, t1,t2)
- Instr(string1,string2)

- **Conversion**

Asc("caractère"): renvoie le nombre auquel il correspond dans table ASCII

Chr(nombre): renvoie le caractère Ascii

Cnum(texte): convertit une chaine en numerique

Cint(nombre): renvoie la partie entiere (en fait convertit en entier)

Cdbl(nombre): convertit un entier en double

Cstr(nombre): convertit un nombre en texte

texte(Chaine,format): renvoie le format defini de la chaine

Equivalent VBA

- Asc("A")=65
- Chr(97)="a"
- Val("24 kg")=24 <> Val("kg 24")=0
- Cint(25.32)=25
- Cdbl(25)=25.00
- Cstr(25)="25"
- Format("22/09/2019", "yyyy\mm\dd")="2019\09\22"

The background features a large, light gray logo for 'ISTP' (Institut Supérieur des Techniciens Professionnels) and the word 'INGÉNIEURS' below it. To the right, there is a stylized, light blue figure of a person with arms raised in a 'V' shape.

FONCTIONS PERSONNALISEES - VBA

Le compilateur repère le début du programme: **la procédure Principale**

appelée par le système d'exploitation

n'existe pas en VBA (≈ evenement Workbook_Open si on veut lancer à l'ouverture du fichier)



Procédure principale suit les instructions: **Déroulement**

si actions répétitives de traitements:
Sous- procédures

si actions répétitives avec attente d'un retour: **Fonctions personnalisées**



Le programme devient **Modulaire**:
Transmission d'**arguments** (paramètres) **typés**

procédures ou fonctions

Retour typé du traitement:
fonctions seulement

- Procédures sans et avec arguments

Procedure NomProcedure()

Instructions

FinProcedure

Procedure NomProcedure(Arg1, [Arg2])

SI Arg1=... ALORS \exemple...

instructions

FINSI

FinProcedure

Exemple

Procedure EliminerLignes()

Pour lig ← 50 à 100

supprimer.ligneEntiere(lig)

FinPour

FinProcedure

Procedure ElimineCertainesLignes(str en string)

Pour lig ← 50 à 100

SI gauche(cellule(lig,2),3)=str

ALORS

supprimer.ligneEntiere(lig)

FinSI

FinPour

FinProcedure

- Fonctions sans et avec arguments

Fonction NomFonction **() en type**

Instructions

FinFonction

Fonction NomFonction(**Arg1**, [**Arg2**]) **en type**

SI **Arg1**=... ALORS \exemple...

instructions

NomFonction ← ...

SINON

NomFonction ← ...

FINSI

FinFonction

Exemple

Fonction AdditionnerPlage() **en long**

Var calc: double

Pour lig ← 50 à 100

calc ← calc + cellule(lig, 1)

FinPour

AdditionnerPlage ← calc

FinFonction

Fonction testDroits(**str en string**) **en bool**

Selon **str**

cas "Admin"

testdroits ← **Vrai**

cas "User"

tesdroits ← **Faux**

FinSelon

FinFonction

• Exo1

- Ecrire un prog qui va afficher un message d'avertissement au moyen d'une sous procedure

```
Debut  
avertissement  
Fin  
procedure avertissement()  
Ecrire "La valeur saisie..."  
FinProcedure
```

• Exo2

- idem précédent mais qui permettra d'afficher un message différent à chaque appel

```
Var Msg: string
Debut
Msg ← "Erreur de..."
avertissement(Msg)
...
Msg ← "La valeur saisie..."
avertissement(Msg)
...
Fin
Procédure avertissement(strMsg: string)
Ecrire strMsg
FinProcédure
```


• Exo3

- Ecrire une fonction qui retournerait la tension d'après les valeurs de Puissance et d'intensité proposées

```
Var P, I, U: Entiers reels  
Debut  
P ← 4400  
I ← 20  
U ← Tension(P,I)  
Fin
```

```
Fonction Tension(intP:Entier, intI:Entier) en Entier  
Si intI=0 ALORS                                     \\si oubli, Erreur!  
    Tension ← 0  
SINON  
    Tension ← intP/intI  
FinSi  
FinFonction
```

• Exo4

- Ecrire une fonction qui vérifierait si une valeur texte appartient à un tableau en affichant vrai ou faux

```
Var saisie: string
Var Tableau String: tab (10)
Debut
tab(0) ← ("test")
tab(...) ← ("...")
tab(9) ← ("essai")
    Ecrire "entrez un texte"
    lire saisie
    Ecrire verifieTexte(saisie,tab)
Fin
```

```
Fonction verifieTexte(strSaisie: string, tablo: variant) en boleen
verifieTexte ← Trouve(strSaisie,joindre(tablo))>0
FinFonction
```

\\joindre: fonction qui concatène tous les éléments d'un tableau; VBA: Join

- Exo

Créer 1 fonction récursive qui permet d'entrer 2 nombres en paramètres, le second étant l'exposant du premier; renvoyer le résultat du calcul depuis la procédure d'appel.

expo(2,5)=32
(2*2*2*2*2)

Algo Appel_exposant

Var Nb, Ex: long

Debut

lire Nb: **lire** Ex

Ecrire **exposant**(Nb,Ex)

Fin

Fonction **exposant**(Nbr:long,Expo:long) en long

Si Expo >=1 **Alors**

exposant=exposant(Nbr,expo-1)*Nbr

Sinon

exposant=1 \\permet de sortir de la pile

FinSi

FinFonction