



Procédures & fonctions

Motivation (1)

2

- Un programme écrit en un seul bloc:
 - ▣ C'est difficile à comprendre
 - ▣ Longueur du code.
- **Solution:**
 - ▣ L'utilisation de sous programmes.
 - ▣ Décomposition du programme en modules élémentaires
- **Avantages:**
 - ▣ Un code plus lisible
 - ▣ Réutilisabilité du code.

Motivation (2)

3

- ***Quelles est l'utilité des sous programme?***
 - ▣ On utilise la notion de sous programmes pour les raisons suivantes:
 - ***Grain du temps (codage)***
 - ***Lisibilité des programmes***
 - ***Gain d'espace mémoire***
 - ***Réutilisabilité des sous programmes sans avoir à les redéclarer***
 - ***Faciliter la maintenance ou la modification des programmes***

Les procédures (1)

4

□ Qu'est ce qu'une procédure ?

- Une procédure est un sous programme, qui peut être appelé dans un autre programme ou dans des différents lieux du programme.
- Une procédure peut être appelée comme une instruction dans un programme

■ Exemple:

If (a<b) **then**

echange (a, b) ;

Les procédures (2)

5

□ Déclaration d'une procédure

```
procedure <nom de la procédure> (liste des paramètres);  
  var <déclaration des variables>  
  begin  
    <Corps de la procédure>  
  end;
```

Non
obligatoire

Exemple

```
procedure echange(a,b: integer);  
Var x: integer;  
begin  
  x:=a;  
  a:=b;  
  b:=x;  
end;
```

Les fonctions (1)

6

□ Qu'est ce qu'une fonction?

- ▣ Une fonction est un cas particulier d'une procédure;
- ▣ Contrairement à une procédure, 1 fct doit avoir un type.
 - c-à-d: celle-ci doit remettre une valeur de sortie.
 - Exemple: `D := delta (a,b,c) ;`

Les fonctions (2)

7

□ Déclaration d'une fonction

```
function <nom de la fonction> (liste des paramètres) :<Type>;  
    var <déclaration des variables>  
    begin  
        <Corps de la fonction>  
    end;
```

Type de la valeur
donnée par la fonction

Exemple

```
function delta(a,b,c: real): real;  
Var d: real;  
begin  
    d:=sqr(b)-4*a*c;  
    delta:=d;  
end;
```

10/02/2019

Variables globales & locales

9

- **Globale:** var déclarée dans le prg principal.
- **Locale:** var déclarée dans une proc ou une fct.
- **Exemple:**

```
program valeur;  
    var x, y: real  
    procedure echange(a,b: integer);  
        Var c: integer;  
        begin  
            a:=x; b:=y;  
            c:=a; a:=b; b:=c;  
            x:=a; y:=b;  
        end;  
    Begin  
        Readln(x, y);  
        echange (x,y);  
    end.
```

Var globale

Var locale

Passage de paramètres

10

- Il existe deux modes de passages de paramètres:
 - ▣ *Le passage par valeur ;*
 - *Recopier la valeur à transférer au sein de la procédure elle-même.*
 - *La procédure travaille avec la copie sans toucher à la valeur d'origine.*
 - ▣ *Le passage par variable (ou par adresse)*
 - *Variable = id + @mémoire.*
 - *Pas de copie de valeur; mais, transmission d'adresses des variables.*

Passage par valeurs - Exemple

11

```
program valeur;  
  var x,y: real;  
  procedure echange (A, B: real);  
    Var C: real;  
  begin  
    A:=x;  
    B:=y;  
    C:=A;  
    A:=B;  
    B:=C;  
    x:=A;  
    y:=B;  
  end;  
begin  
  echange (x,y);  
end.
```

Vars globale

Paramètres **sans** le mot clé Var

Passage des valeurs des
variables globales

Permutation des valeurs

Restitution des valeurs après
permutation

Appel de la procédure

Passage par adresses - Exemple

12

```
program valeur;  
  var x,y: real;  
  procedure echange (var A, B: real);  
    Var C: real;  
  begin  
    C:=A;  
    A:=B;  
    B:=C;  
  end;  
begin  
  echange (x,y);  
end.
```

Vars globale

Paramètres **avec** le mot clé Var

Permutation des valeurs

Appel de la procédure

Série de TD sur les sous_pgmes

13

□ Exercice 01

- Proposer un programme permettant de :
 - 1) Saisir deux entiers A et B.
 - 2) Appeler une procédure pour calculer la somme de A et B
 - 3) Appeler une fonction pour calculer le produit de A et B.

Série de TD sur les sous_pgmes

14

Exercice 02

Programme permettant de :

- 1) lire un entier N .
- 2) appeler une fonction pour calculer la somme des entiers de 1 à N .

Série de TD sur les sous_pgmes

15

□ Exercice 03

- Un programme qui appelle :
- Une fonction pour calculer la puissance x^Y
- Une procédure pour convertir un nombre binaire en décimal.