

# **Fonctions et procédures**

**NF01**

**Philippe TRIGANO**



# Fonctions standard

- `sqr(x)` : carré de `x`
- `sqrt(x)` : racine carrée
- `sin(x)`, `cos(x)`, ... : fonctions trigonométriques
- `chr(code)` : retourne le caractère dont le code ASCII est `x`
- `ord(car)` : retourne le code correspondant au caractère `car`

## Exemples d'utilisation

```
delta := sqr(b) - 4 * a * c;
```

```
x1 := (-b + sqrt(delta)) / (2 * a);
```

# Exemple introductif

- **On dispose de données sur un ensemble d'individus : taille, poids, âge, ...**
  
- **Comment calculer :**
  - la taille moyenne,
  - le poids moyen,
  - l'âge moyen ?

# Solutions

## ➤ 1<sup>ère</sup> solution

- Calcul de la somme des tailles
- Division du résultat par le nombre d'individus
- Calcul de la somme des poids
- Division du résultat par le nombre d'individus
- Calcul de la somme des âges
- Division du résultat par le nombre d'individus

## ➤ Une meilleure solution ?

- Ecrire une **fonction** ou une **procédure** permettant de calculer la moyenne de n nombres
- Passer les valeurs des tailles, poids et âges en **paramètres**



# Objectifs

- **Eviter la répétition d'instructions**  
correspondant à des traitements analogues
- **Structurer les programmes**
  - Exemple :
    - ✓ lecture
    - ✓ calcul
    - ✓ affichage
- **Une procédure permet de définir un traitement autonome**
  - nommé par un identificateur
  - appelable par cet identificateur
- **Une fonction retourne de plus une valeur**



# Exemple de procédure

## ➤ Affichage des coordonnées d'un vecteur

**type**

Vecteur = array[1..Nmax] of real ;

Paramètres  
(typés)

**procedure** affichage ((v): Vecteur, (n): integer) ;

**var**

i: integer;

**begin**

for i:=1 to n do write(v[i]);

writeln;

**end;**



# Appel d'une procédure

- On écrit simplement le nom de la procédure suivi des paramètres d'appel

- Exemple

```
v1[1]:= 5;
```

```
v1[2]:=-1;
```

```
v1[3]:= 3;
```

```
affichage(v1 , 3);
```

## Exemple de Fonction :

```
function moyenne(x, y, z : real) : real;
```

```
begin
```

```
    moyenne := (x+y+z) / 3;
```

```
end;
```



# Appel d'une fonction

➤ **Au niveau d'une **expression****

(et non d'une instruction comme pour une procédure)

➤ **Exemple :**

$x := \text{cube}(y); \quad z := 3 * \text{cube}(t) + 1;$

➤ **Remarques**

- La fonction est appelée lors de l'évaluation de l'expression
- La valeur qu'elle retourne est utilisée dans l'évaluation





# Exemple complet

```
program Puissance3 ;
```

```
var
```

```
    unNombre : real;
```

```
function cube (x : real) : real;
```

```
begin
```

```
    cube:=x*x*x;
```

```
end;                { Fin du code de la fonction }
```

**Déclarations :**

- Une variable globale
- Une fonction

**Corps du programme**

```
begin
```

```
    readln (unNombre);
```

```
    writeln ('Le cube de', unNombre,' est : ', cube(unNombre) ;
```

```
end.
```

