

5. Les procédures et les fonctions

Elles permettent de réaliser une tâche :
action plus ou moins complexe sur une ou
plusieurs variables

Les procédures

- ❑ Nouvelles **instructions** créées à l'initiative du programmeur et identifiées par un verbe symbolisant l'action.
- ❑ Chaque donnée et chaque résultat doivent être identifiés.
- ❑ Spécifier une procédure c'est identifier
 - l'action réalisée par cette procédure,
 - les *paramètres formels* (*servant à écrire la procédure*)
 - + l'ensemble des données nécessaires avec leur type,
 - + l'ensemble des résultats produits avec leur type.et décrire l'état des données et des résultats avant et après exécution de la procédure.

Format

Procédure nom-proc (E variables avec types ; E/S variables avec types ; S variables avec types)

E : paramètre de donnée, ne subit aucune altération au cours de l'exécution de la procédure (pas d'affectation de ce paramètre dans la procédure).

S : paramètre résultat de l'action, sa valeur n'est pas significative avant l'exécution de la procédure (initialisation obligatoire de ce paramètre dans la procédure).

E/S : paramètre de donnée-résultat, sert de donnée et de résultat.

Exemple : échange de deux variables.

Procédure échanger (E/S A, B : entier)

Var C : entier

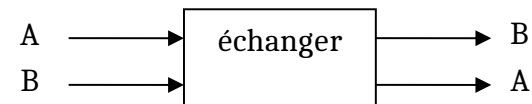
Début

C ← A

A ← B

B ← C

Fin



Appel

- ❑ Les procédures sont placées avant le début des instructions et après la déclaration des variables.
- ❑ Pour appeler une procédure, il suffit de donner son identificateur avec les variables (paramètres effectifs) sur lesquelles l'action va s'exercer.
- ❑ Contraintes à suivre :
 - même nombre de paramètres effectifs que de paramètres formels,
 - les paramètres effectifs sont associés aux paramètres formels dans leur ordre,
 - les paramètres effectifs sont du même type que les paramètres formels associés,

Exemple : appel de la procédure échanger

```
Var C, D, E : entier  
C ← 2  
D ← 3  
échanger (C, D)  
E ← 5  
échanger (E, C)
```

Les fonctions

- ❑ Même sens qu'en mathématique.
- ❑ On utilise une fonction f pour obtenir la valeur $f(x)$ pour une valeur x .
Un seul résultat.
procédure : instruction
fonction : expression
- ❑ Dans une fonction, les paramètres formels sont par définition des paramètres en entrée.

Fonction nom-fonc (param. en entrée avec types) : type du résultat

Exemple : pour un tableau t , écrire une fonction pos-min qui donne la position du minimum entre deux indices i et j .

Fonction pos-min (t : tableau ; i, j : entier) : entier

Exemple : pos-min

```
Fonction pos-min (t : tableau ; i, j : entier) : entier
Var k, l : entier
Début
k ← i
Pour l ← i+1 à j inc +1 Faire
    Si t[l] < t[k]
        Alors k ← l
    FinSi
FinPour
retourne (k)
Fin
```

Visibilité des objets

- ❑ Pour un module (procédure ou fonction)
 - les *objets locaux* sont définis dans le module, ils ne peuvent pas être utilisés en dehors,
 - les *objets globaux* sont définis en dehors du module.
- ❑ Des objets globaux ne peuvent pas porter le même nom.
- ❑ Les objets locaux à un module ne peuvent pas porter le même nom.
- ❑ Des objets locaux à différents modules peuvent porter le même nom.
- ❑ Un objet local à un module peut porter le même nom qu'un objet global :
 - dans le module, ce nom désigne l'objet local
 - à l'extérieur du module, il désigne l'objet global.