

# Elements d'algorithmique

## TP 4 : les tableaux (1)

### 1 Motivation

En programmation, pour représenter un ensemble fini de données du même type (exemple : les coordonnées d'un point dans le plan, les notes des élèves d'une classe, les jours de la semaine, ...), on utilise des tableaux.

### 2 Déclaration d'un tableau

#### 2.1 Déclaration simple

Canevas de déclaration simple d'un tableau :

```
VAR nom_tableau : ARRAY [1..taille] OF type;
```

Exemple :

```
VAR v1 : ARRAY [1..3] OF REAL;  
VAR n1 : ARRAY [1..30] OF INTEGER;  
VAR s1 : ARRAY [1..7] OF STRING;
```

#### 2.2 Déclaration de type

Canevas de déclaration d'un tableau grâce à un type :

```
TYPE nom_type = ARRAY [1..taille] OF type;  
VAR nom_variable : nom_type;
```

Exemple :

```
TYPE vecteur = ARRAY [1..3] OF REAL;  
VAR v1 : vecteur;
```

### 3 Accès aux éléments d'un tableau

Les éléments d'un tableau  $t$  sont notés  $t[1]$ ,  $t[2]$ ,  $t[3]$ , ... (à l'aide d'indices entre crochets). L'élément  $t[i]$  est une variable manipulable comme n'importe quelle variable. On peut en particulier utiliser l'affectation, *readln*, *writeln* pour modifier ou afficher sa valeur.

Exemple :

```
PROGRAM exemple;

VAR date : ARRAY[1..3] OF INTEGER;

BEGIN
date[1] := 15;
date[2] := 10;
date[3] := 1981;
writeln('Je suis né le ',date[1], '/', date[2], ' en ',date[3]);
readln;
END.
```

### 4 Exercices

#### 4.1 Exercice 1

Ecrire un programme qui demande à l'utilisateur d'entrer un entier  $N$  puis demande à l'utilisateur de saisir  $N$  entiers qui seront placés dans un tableau et enfin, affiche ces  $N$  nombres séparés par des virgules.

#### 4.2 Exercice 2

Ecrire un programme affichant le produit scalaire et la somme de deux vecteurs dans le plan.

#### 4.3 Exercice 3

Ecrire un programme calculant la moyenne, la note minimale, la note maximale d'un tableau de notes (càd, un entier entre 1 et 20) saisies auparavant par l'utilisateur.