

Les Enregistrements

HAMID NECIR | COURS D'ALGORITHMIQUE

Introduction

2

- ▶ Toute donnée élémentaire vue précédemment en cours, doit avoir **un seul type** (entier réel, caractère,...).
- ▶ Meme et les chaines et les tableaux qui regroupent **plusieurs données doivent être toutes de même type**.
- ▶ Exemple : var x: entier;
- ▶ T: tableau [1..50] de réels;
- ▶ C: chaine;

Inconvénient

3

ces structures ne répondent pas à nos besoins si nous voulons regrouper des données de différents types.

► **Exemple :** Fiche d'un étudiant

- | | | |
|---|-------------|-----------|
| ► | - matricule | entier |
| ► | - nom | chaîne |
| ► | - prénom | chaîne |
| ► | - âge, | entier |
| ► | - sexe | caractère |

Ces variables sont de différents types

Déclaration

4

- Un enregistrement est une structure de données dont **les éléments peuvent être de type différent**. Les éléments qui composent un enregistrement sont appelés **champs**.

Syntaxe :

```
Var <Nom enregistrement>= enregistrement
    < Nom_champ_1> : <type_champ_1>;
    < Nom_champ_2> : <type_champ_2>;
    ...
    < Nom_champ_N> : <type_champ_N>;

FinEnreg;
```

Exemple

5

► En tant que type:

Type etud= enregistrement

matricule: entier;

nom, prénom: chaîne[20];

âge: entier;

sexe: caractère;

Finenreg;

Var Étudiant: etud;

Exemple

6

b) **En tant que variable:**

Var Étudiant : enregistrement

matricule: entier;

nom, prénom: chaîne[20];

age: entier;

sexe: caractère;

Finenreg;

Exemple

7

b) En tant que variable:

Var Étudiant = enregistrement

matricule: entier;

nom, prénom: chaîne[20];

âge: entier;

sexe: caractère;

Finenreg;

Manipulation d'un enregistrement

8

- ▶ La manipulation d'un enregistrement ne se fait pas globalement mais seulement au travers de ses champs.
- ▶ il n'est pas possible de manipuler un enregistrement, sauf pour affecter un enregistrement à un autre de même type.
- ▶ Par exemple, pour afficher un enregistrement il faut afficher tous ses champs uns par uns.

Accès aux champs

9

Pour accéder à un membre d'un enregistrement, on procède comme suit :

Syntaxe :

<Nom_enregistrement>. <Nom_champ>;

Exemple

accéder à un champ de la variable étudiant, on procède comme suit:

Etudiant.matricule;

Etudiant.nom;

Etudiant.prénom;

Etudiant.âge ;

Structure Avec .. Faire

10

Pour simplifier l'écriture et éviter l'utilisation répétée des champs et de la notation avec le point (Nom enregistrement.champ), on peut utiliser l'instruction Avec .. Faire.

Cette structure s'utilise aussi bien avec une opération d'affectation, de lecture ou d'écriture.

Exemple

Avec Etudiant Faire

matricule ← 5201800021;

Lire(nom);

Lire(prénom);

Écrire(matricule);

Fait;

Affectation

11

L'affectation se fait champs par champ, on procède comme suit :

Syntaxe

1ière méthode :	2ème méthode :
<code><Nom_enregistrement>.<Nom_champ>←valeur;</code>	<code>avec <Nom_enregistrement > faire NomChamp ← valeur; Fait;</code>

Exemple: Pour affecter des valeurs un enregistrement Etudiant, on procède comme suit:

Sans l'instruction Avec	Avec l'instruction Avec
<code>Etudiant.matricule ← 5201800021; Etudiant.nom ← "Mohamedî"; Etudiant.prénom ← "omar"; Etudiant.sexe ← "M"; Etudiant.âge ← 18;</code>	<code>avec Etudiant faire matricule ← 5201800021; nom ← "Mohamedî"; prénom ← "omar"; sexe ← "M"; âge ← 18; Fait;</code>

Affectation

12

L'affectation d'un enregistrement ne se fait globalement que pour affecter un enregistrement à un autre de même type.

► Exemple

Type Livre= enregistrement

ISBN: entier;

Titre: chaine[40];

NbPage: entier;

Finenreg;

Var bouquin, manuel: Livre;

bouquin ← manuel;

Instruction de lecture

13

La lecture ou d'écriture d'un enregistrement se fait champ par champ.

Syntaxe

1ère méthode :	2ème méthode :
<code>Lire(<Nom_enregistrement>. <Nom_champ>);</code>	<code>avec <Nom_enregistrement> faire Lire(Nom_Champ); Fin avec;</code>

Sans l'instruction Avec	Avec l'instruction Avec
<code>Lire(Etudiant.matricule); Lire(Etudiant.nom); Lire(Etudiant.prénom); Lire(Etudiant.sexe); Lire(Etudiant.âge);</code>	<code>avec Etudiant faire Lire(matricule); Lire(nom); Lire(prénom); Lire(sexe); Lire(âge); Fait;</code>

Instruction d'écriture

14

L'écriture d'un enregistrement se fait champ par champ.

Syntaxe

Sans l'instruction Avec	Avec l'instruction Avec
Écrire(<Nom_enregistrement>.<Nom_champ>);	avec <Nom_enregistrement> faire Écrire(Nom_Champ); Fin avec;

Exemple

Sans l'instruction Avec	Avec l'instruction Avec
Écrire(Etudiant.matricule); Écrire(Etudiant.nom); Écrire(Etudiant.prénom); Écrire(Etudiant.sexe); Écrire(Etudiant.âge);	avec Etudiant faire Écrire(matricule); Écrire(nom); Écrire(prénom); Écrire(sexe); Écrire(âge); Fait;

Enregistrement d'enregistrements

15

C'est le cas où un ou plusieurs champs de l'enregistrement sont des enregistrements.

Exemple

```
Type Date= enregistrement  
    Jour : 1..31;  
    mois : 1..12;  
    année: 1990...2060;  
Finenreg;
```

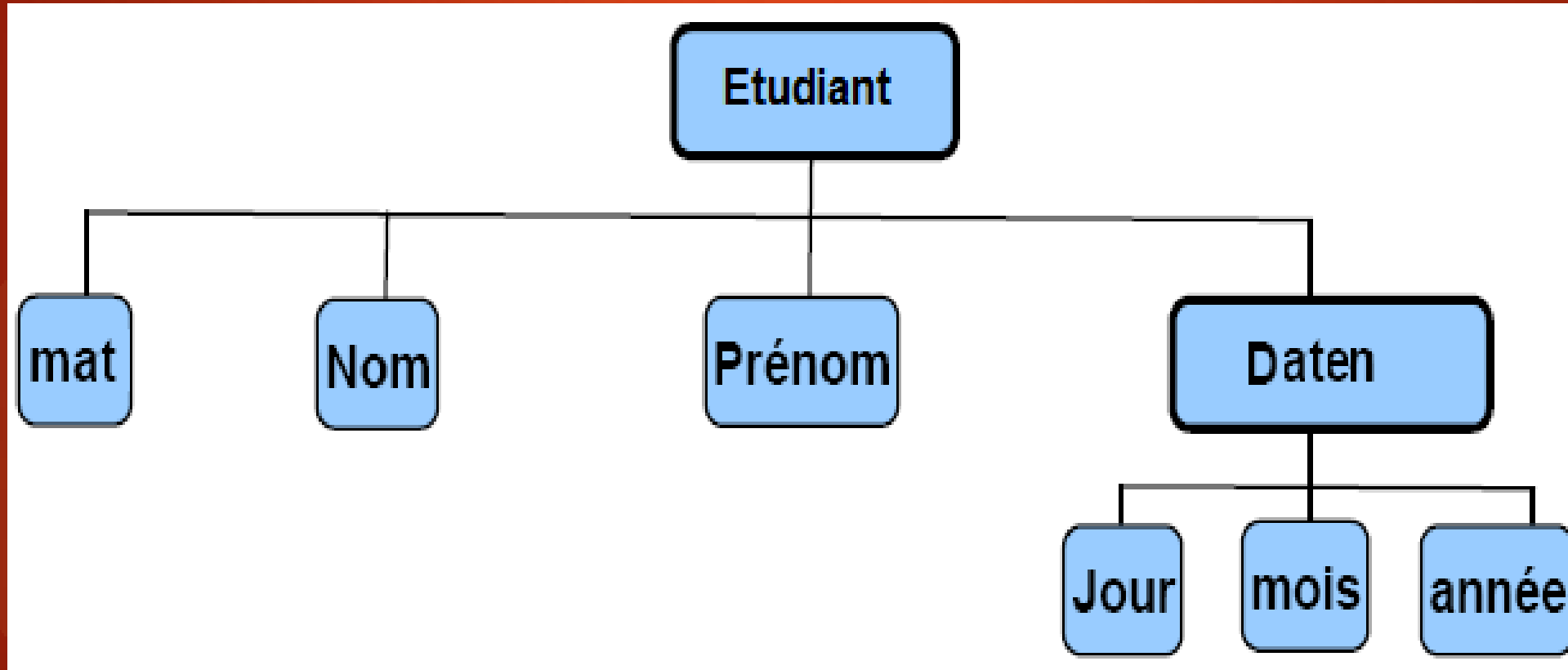
```
etud= enregistrement  
    mat: entier;  
    nom, prénom: chaîne[20];  
    Date: Date;  
Finenreg;
```

```
Var Etudiant: etud;
```

Représentation schématique

16

Cette déclaration peut être représentée par un arbre comme nous montre le schéma suivant :



Remarque

17

Dans le cas d'un enregistrement imbriqué, on utilisera plusieurs instructions

Avec imbriquées.

Exemple:

avec Etudiant faire

Lire(mat);

Lire(nom);

Lire(prénom);

avec Daten faire

Lire(jour);

Lire(mois);

Lire(annee);

Fait;

Fait;

Tableaux d'enregistrements :

18

- Il arrive souvent que l'on a à traiter non pas un seul enregistrement mais plusieurs. Par exemple, on veut traiter une section d'étudiants. Donc on ne va pas créer autant de variables de types étudiants (enregistrement) qu'il y a d'étudiants.

Exemple:

Ecrire un algorithme permettant de saisir les informations de l'ensemble des étudiants d'une section contenant N étudiants ($N \leq 180$). Par la suite afficher les étudiants du groupe 2 qui ont une note supérieure ou égale à 10.

Tableaux d'enregistrements :

19

algorithme Exercice;

type etud = enregistrement

 nom, prénom : chaine[20] ;

 Matricule: entier ;

 groupe : 1..4;

 note : 0..20;

 fait;

var Section: tableau[1,...,180] de étud ;

 i, N : entier;

Tableaux d'enregistrements :

20

début

Répéter lire(N) ; jusqu'à (N \geq 1) et (N \leq 180);

/*Lecture du tableau section*/

pour i \leftarrow 1 à N faire

avec Section[i] faire

lire(Matricule) ;

lire (nom, prénom) ;

lire (groupe) ; lire (note) ;

fait ;

fait ;

Suite algorithme:

21

pour $i \leftarrow 1$ à N faire

si (Section[i].groupe = 2) et (Section[i].note \geq 10) **alors**

 écrire (Section[i].nom, Section[i].prénom);

fsi ;

fait ;

fin.

Enregistrement de tableaux :

22

Un enregistrement de tableaux est un enregistrement dont un ou plusieurs champs sont de type tableau.

Exercice :

Chaque étudiant est désigné par son nom et prénom (chaîne [20]), numéro de groupe (1..4), et ses notes obtenues dans 8 modules (algorithmique, analyse, algèbre, mécanique et bureautique,...).

Ecrire un algorithme qui permet de lire les enregistrements des 160 étudiants de la première année LMD et d'afficher les noms et prénom des étudiants admis en 2ème année (moyenne ≥ 10).

Les coefficients des modules sont 1.

23

On peut schematiser notre section comme suit:

The diagram illustrates a database schema for a course evaluation system. It features two main sections, SECTION[1] and SECTION[2], each containing a table of student evaluations. The attributes for each section are: matricule, nom, prenom, groupe, and note. The data is organized into a table with rows for DEVOIR, TP, and EMD, and columns for CHIMIE, PHYSIQUE, ALGO, and ANALYSE.

SECTION[1]					SECTION[2]					
matricule	nom	prenom	groupe	note	matricule	nom	prenom	groupe	note	
										DEVOIR
										TP
										EMD
										CHIMIE PHYSIQUE ALGO ANALYSE

Exercice :

24

Algorithme Exercice ;

Type etudiant = enregistrement

 Matricule: entier ;

 nom, prénom : chaine[20] ;

 groupe : 1..4 ;

 note : tableau [1..3,1..4] de réel; /* 3 notes(dev, TP, EMD) pour chacun des 4 modules

finEnreg ;

Var section= tab [1..160] de etudiant ;

 i, j, : entier ;

 moy, s, ST : réel ;

début

pour $i \leftarrow 1$ **a** n **faire**

avec Section[i] **faire**

Lire(Matricule); lire(nom, prénom); lire (groupe);

$ST \leftarrow 0$;

pour $k \leftarrow 1$ **a** 4 **faire**

$S \leftarrow 0$;

pour $j \leftarrow 1$ **a** 3 **faire**

lire (note[j,k]);

$S \leftarrow S + \text{note}[j,k]$;

fait ;

$\text{moy} \leftarrow S/3$;

$ST \leftarrow ST + \text{moy}$;

fait ;

$\text{MoyG} \leftarrow ST/4$;

/* Moyenne d'un module

/* Moyenne générale

```
si (MoyG >= 10) alors écrire(" étudiant admis", Section[i].nom, Section[i].prénom) ;  
    Sinon écrire(" étudiant non admis", Section[i].nom, Section[i].prénom) ;  
  
    fsi ;  
fait;  
fait ;  
Fin.
```

Enregistrement paramètre d'une action paramétrée

27

Un enregistrement est passer complètement en paramètre d'une fonction ou d'une procédure (on n'est pas obligé de passer tous les champs uns à uns).

Exemple :

Ecrire une fonction qui renvoie la différence d'âge entre deux personnes

Exemple

```
Type tpersonne=Enregistrement
```

```
    Nom, prénom : chaine[20];
```

```
    âge : entier;
```

```
Finenreg;
```

```
Var pers1, pers2: tpersonne;
```

Exemple :

28

Fonction différence (E/p1, p2 :tpersonne):entier;

Début

Si p1.age > p2.age **Alors**

différence ← p1.age – p2.age;

Sinon différence ← p2.age – p1.age;

Fsi;

Fin;