

Les enregistrements

Synthese de la définition d'un enregistrement :

Nom Enreg : Enregistrement

NormRamp1: type Ramp1

Norm Champ2: type Champ2

Nom Champ : type Champ

Fin Enreg

Exp:

Date: Enregistrement

J, M, A : entier

Fin Enreg

Etudiant: Enregistrement

Id: entier

Nom : Graïne

DN: Date

Notes: tableau [1..10] de réel

May: need

Fin Enreg

Rq: La définition (creation) d'un enreg
c'est l'équivalent de la creation d'un
nouveau type.

Declaration de variable de type enregistrement:

Simple

Nom Var: Nom E neg

D1: Date \Rightarrow D1

E1: Etudiante

Composée

NomTab: Tableau

[ind...inf] de nom
enregistrement

Exp:

Group: Tableau

[1..50] d'étudiant

Var Enreg ← Var Enreg

⇒ L'utilisation d'un champ de type Enregistrement se faire généralement champ par champ.

L'accès aux champs :

Nom var. Nom Gramp

Exp:

D1 J → D1.J
M → D1.M
A → D1.A

Diagram illustrating the structure of a record (E1) with fields:

- Id
- Norm
- DN
- Note
- May

Each field is connected to its corresponding label on the right:

- Id → E1.Id
- Norm → E1.&Norm
- DN → E1.DN.A
- Note → E1.Notes[i]

Groupe: tableau d'étudiants

Groupe[i].Ed

groupe[i].DN.A

Grp[i].Note[j]

utilisation des enregistrement avec les fonctions / procédures

procédure Nomproc (paramètre: NomEntree)

Exercice : donner la SD ~~adéquate~~ pour représenter la deg d'un polynôme.

- 1) Définir une SD permettant de représenter un monôme.
- 2) Définir une SD permettant de représenter un polynôme composé de N monômes.
- 3) Ecrire les fonctions/proc suivants :
 - saisir d'un monôme
 - saisir d'un polynôme
 - déterminer l'image d'une valeur X données selon un polynôme donné.
 - déterminer la dérivée d'un polynôme donné.

correction :

0) la SD ~~adéquate~~ ~~est~~ pour représenter un deg est un variable simple de type entier

1) un ~~mot~~ monôme peut être représenter
un monôme est un enregistrement

~~coef~~

monôme : enregistrement

coef : réel

puiss : entier

fin enreg

Ch 2 - Les pointeurs

un pointeur est une variable spéciale capable de recevoir l'adresse d'une autre variable d'un type donné.

Déclaration d'un pointeur :

$\text{Nomptr} : * \text{Type}$

Manipulation des pointeurs :

$P : * \text{Entier}$

$A : \text{Entier}$

$P \leftarrow \text{Null}$

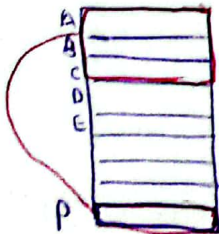


$P \leftarrow \&A$



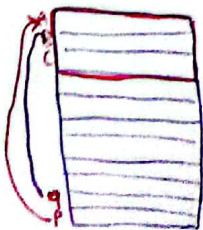
$P \leftarrow \text{Allouer}(n)$

Exp: $p \leftarrow \text{Allouer}(3)$



$q : * \text{Entier}$

$q \leftarrow *p$



$q \leftarrow p + n$

Exp

$q \leftarrow p + 1$

$m \leftarrow q - p$

Libération d'un couple référence par un ptr :

$\text{liberer}(\text{ptr})$

Exp

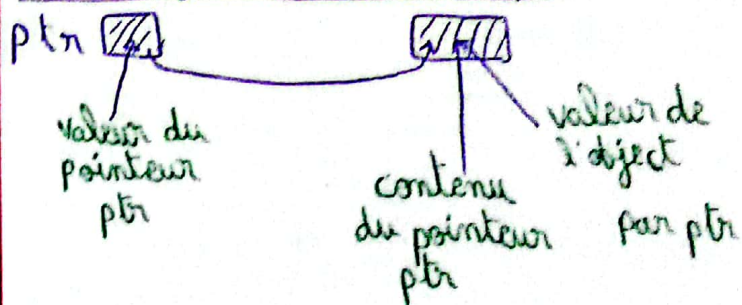
$A : \text{Entier}$

$p, q : * \text{Entier}$

$p \leftarrow \&A$

$q \leftarrow \text{Allouer}(4)$

Accès au contenu d'un ptr :



Le contenu d'un ptr est identifié par $* \text{ptr}$

Exp

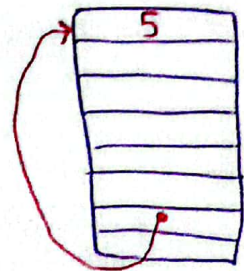
$A : \text{Entier}$

$P : * \text{Entier}$

$A \leftarrow 5$

$P \leftarrow \&A$

$\text{Ecrire}(A) \Rightarrow \text{Ecrire}(*P)$



Les pointeurs et les proc / func :

1) le résultat d'une fonction peut être une pointeur

2) le passage par variable peut être remplacé par le passage d'un ptr.

~~pas point~~

proc Permutation (var A : entier, B : entier)

var x ← Entier

Debut

x ← A

A ← B

B ← x


```

Proc permutation (A: *Entier, B: *entier)
  Var
    x: entier
  Debut
    x ← *A
    *A ← *B
    *B ← x
  Fin

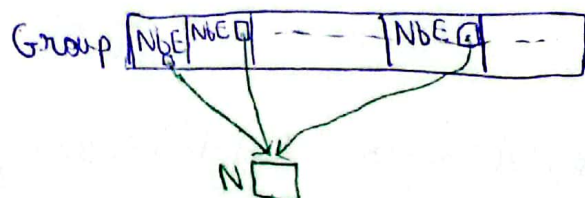
```

Les pointeurs et les enregistrements

1) Etudiant: enregistrement

Id: entier
NbE: Entier

Fin Enreg



un champ dans un enreg peut être de type pointeur

Exp

Etudiant: Enregistrement

Id: entier

NbE: *entier

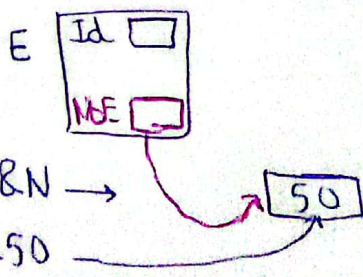
fin Enreg

~~Var Enreg: Nom Enreg~~

Var Enreg: Nom Enreg
Var Enreg.Nom Champ
*(Var Enreg.Nom Champ)

E: Etudiant

N: entier



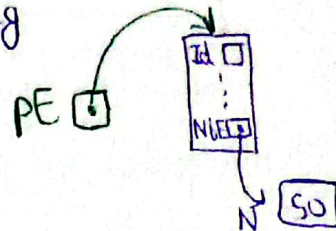
2) un variable de type enreg peut être référencé par un pointeur.

ptr Enreg: #Nom Enreg

Exp

PE: * Etudiant

ptr Enreg ← &Var Enreg



Exp:

PE ← &E

Var Enreg.Nom Champ ↔ *ptr Enreg.Nom Champ

Exp:

E.Id ← 50 ↔ PE → Id ← 50

ptr Enreg → Nom Champ

*(PE → NbE) ← 60

Les pointeurs et les tableaux

un tableau est un pointeur constant (statique)

T: *type

N: Entier

lire(N)

T ← Allouer (N)

Exp:

Algorithme scolaire

type

Date: enregistrement

J, M, A: entier

Fin Enreg

Etudiant: enregistrement

Id: entier

Nom: chaîne

DN: Date

Note : * réel
NbE : * entier

fin enreg

var

N, M : entier
T : * Etudiant
P : * Etudiant
q : * réel

debut

Ecrire ("donner le nombre des étudiants")

lire (N)

Ecrire ("donner nb")

lire (M)

T ← Allouer (N)

pour p de T à T+N-1

lire (p → Id)

lire (p → Nom)

lire (p → DN, J, p → DN, M,
p → DN, A)

P → Notes ← Allouer (M)

pour q de p → Note à p → Note + M - 1
lire (*q)

fin pour

P → NbE ← &N

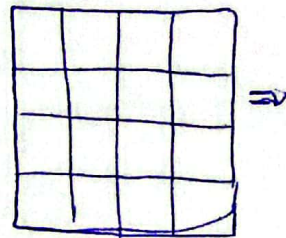
fin pour



pour p de T à T+N-1 faire

Matrice : tableau [1..10][1..10] d'entier

M : Matrice



⇒ M : tableau [1..10] de * entier

lire (b)

lire (c)

M ← Allouer (L)

pour p de M à M+L-1 faire

*p ← Allouer (c)

pour q de *pp à *p+L-1 faire
lire (*q)

fin pour

fin pour