

## Algorithmique - T.D. N° 9

12 décembre 2008

### Les Structures et les listes linéaires

#### 1 Traitement des structures

Soit le fichier "toto" contenant un ensemble de données décrivant l'ensemble des élèves de la première année. Le type élève étant une structure appelée `eleveEisti` contenant plusieurs champs. Nous souhaitons créer un nouveau fichier contenant seulement les champs : nom, prenom, origine pour chaque élève.

**Exercice 1** Écrire un procédure qui charge les données souhaitées dans une structure convenable. Pour simplifier nous supposons que le nombre d'élèves est donné.

**Corrigé 1** type `eleve` = structure

- nom : chaîne
- prenom : chaîne
- origine : chaîne

---

**Algorithme 1** Charger le fichier dans un tableau

---

**procédure** charger (S tableau tab () : `eleve`, E taille : entier)

**variables** f : flot de `eleve`

x : `eleveEisti`

créerTableau(tab,taille)

ouvrir(f,"toto")

i ← 1

**tantque** non eof(f) **faire**

lire(f,x)

tab[i].nom ← x.nom

tab[i].prenom ← x.prenom

tab[i].origine ← x.origine

i ← i + 1

**fin tantque**

fermer(f)

---

**Exercice 2** Écrire une procédure qui permet de créer le nouveau fichier dans lequel on enregistre les données souhaitées.

---

**Algorithme 2** Charger le fichier dans un tableau

---

**procedure** enregistrer (E tableau tab (N) : eleve, E taille : entier)

**variables** f : flot de eleve

  i : entier

  nom : chaine

$i \leftarrow 1$

  ecrire ("entrer le nom du fichier")

  lire(nom)

  ouvrir(f, nom)

**tantque**  $i \leq$  taille **faire**

    ecrire(f, tab[i])

$i \leftarrow i + 1$

**fin tantque**

  fermer(f)

---

**Corrigé 2**

## 2 Traitement des matrice creuses

Nous souhaitons traiter des matrices creuses. Une représentation adaptée serait par une liste linéaire de listes linéaires. La première liste représente l'ensemble des lignes non nulles placées dans l'ordre. Une cellule ligne contient le numéro de ligne et un lien vers la liste des cellules colonnes non nulles, ainsi qu'un autre lien vers la cellule ligne suivante.

Une liste de colonnes associée à une ligne serait une liste linéaire de toutes les cases non nulles dans l'ordre des colonnes.

**Exercice 3** Trouver les structures adaptées qui permettent d'effectuer cette représentation.

**Corrigé 3** type celluleColonne = structure

  - numColonne : entier

  - contenu : entier

  - suivant : celluleColonne

type celluleLigne = structure

  - maLigne : entier

  - contenu : celluleColonne

  - suivant : celluleLigne

**Exercice 4** Soit m une matrice d'entiers représentée ainsi, écrire une fonction permettant de récupérer la valeur de la case  $m(i,j)$ .

**Corrigé 4**

**Exercice 5** Supposons que nous avons deux matrice m1, m2 stockées selon la représentation précédente. Créer une nouvelle matrice qui représente la somme des deux matrices. Pour simplifier on suppose que les matrices sont carrées et de taille *taille*.

**Corrigé5**

---

**Algorithme 3** Récupere m(i,j)

---

**fonction** case (E m : celluleLigne de ligne, E i,j :entier) : entier

**variables** l :celluleLigne

x : entier

trouve : booleen

c :celluleColonne

*trouve* ← FAUX

*l* ← m

**tantque** *l* <> nil ET non trouve **faire**

**si** l.maLigne=i **alors**

*trouve* ← VRAI

**sinon**

*l* ← *l.suivant*

**fin si**

**fin tantque**

**si** trouve **alors**

*trouve* ← FAUX

*c* ← *l.contenu*

**tantque** *c* <> nil ET non trouve **faire**

**si** c.numColonne=j **alors**

*trouve* ← VRAI

**sinon**

*c* ← *c.suivant*

**fin si**

**fin tantque**

**fin si**

**si** trouve **alors**

*x* ← *c.contenu*

**sinon**

*x* ← 0

**fin si**

**retourner** x

---

---

**Algorithme 4** Construire une ligne de résultats

---

**procedure** construireLigne (E m1,m2 : celluleLigne de ligne, E taille, i :entier, S c :celluleColonne de entier)

**variables** l1,l2 :celluleLigne

c1,c2 :celluleColonne

$l1 \leftarrow m1$

$l1 \leftarrow m1$

$c \leftarrow nil$

$c1 \leftarrow nil$

$j \leftarrow 1$

**tantque**  $j \leq \text{taille}$  **faire**

**si**  $\text{case}(m1,i,j) + \text{case}(m2,i,j) \neq 0$  **alors**

*nouveau*(c2)

$c2.\text{contenu} \leftarrow \text{case}(m1, i, j) + \text{case}(m2, i, j)$

$c2.\text{numColonne} \leftarrow j$

**si**  $c1 \neq nil$  **alors**

$c1.\text{suivant} \leftarrow c2$

$c1 \leftarrow c1.\text{suivant}$

**sinon**

$c \leftarrow c2$

$c1 \leftarrow c2$

**fin si**

**fin si**

$j \leftarrow j + 1$

**fin tantque**

---

---

**Algorithme 5** Construire la matrice des résultats

---

**procedure** construireMatrice (E m1,m2 : celluleLigne de ligne, E taille :entier S m :celluleLigne de entier)

**variables** i : entier

l2,l1 :celluleLigne

c : celluleColonne

$m \leftarrow nil$

$l1 \leftarrow nil$

$i \leftarrow 1$

**tantque**  $i \leq \text{taille}$  **faire**

*construireLigne*(m1, m2, taille, i, c)

**si**  $c \neq nil$  **alors**

*nouveau*(l2)

$l2.\text{contenu} \leftarrow c$

$l2.\text{maLigne} \leftarrow i$

**si**  $l1 \neq nil$  **alors**

$l1.\text{suivant} \leftarrow c2$

$l1 \leftarrow l1.\text{suivant}$

**sinon**

$m \leftarrow l2$

$l1 \leftarrow l2$

**fin si**

**fin si**

$i \leftarrow i + 1$

**fin tantque**

---