Tableaux à deux dimensions et boucles imbriquées

DEUG MIAS — IF121 — TP 12

20 janvier 2002

1 Relevés de notes

On étoffe l'exemple du cours (transparents 6 et suivants). Il s'agit de stocker dans un tableau à deux dimensions les notes d'étudiants dans différentes matières.

On définit deux constantes : le nombre de matières et le nombre d'étudiants.

```
const int MATIERES = 3;
const int EFFECTIF = 5;
```

Définir un type `releve` pouvant contenir les notes de tous les étudiants dans toutes les matières. Si `notes` est une variable de type `releve`, `notes[e][m]` est la note de l'étudiant `e` dans la matière `m`.

1.1 Saisie

Écrire une procédure de saisie des notes.

```
void saisie_notes(releve notes);
```

1.2 Moyennes

Écrire une fonction qui calcule la moyenne d'un étudiant (toutes les matières ont le coefficient 1).

```
float moyenne_etudiant(releve notes, int e);
```

Écrire une fonction qui calcule la moyenne de la classe dans une matière.

```
float moyenne_matiere(releve notes, int m);
```

1.3 Moyenne générale

Écrire une fonction qui calcule la moyenne générale de la classe pour toutes les matières réunies.

```
float moyenne_generale(releve notes);
```

1.4 Affichage

Écrire une procédure qui affiche un relevé de notes de la classe. On placera chaque matière dans une colonne, plus une colonne à droite pour la moyenne ; et chaque étudiant sur une ligne, plus une ligne en bas pour la moyenne de classe.

```
float affiche_releve_avec_moyennes(releve notes);
```

1.5 Tracas administratifs

Les notes ont été entrées dans l'ordre (mathématiques, informatique, physique). Mais l'administration demande un relevé de notes dans l'ordre (mathématiques, physique, informatique). Écrire une procédure qui fait l'échange des deux colonnes.

```
echange_matieres(releve notes, int m1, int m2);
```

2 Recherche d'une sous-chaîne

(Pour les plus rapides)

On veut écrire un programme qui recherche un sous-mot dans un mot. Voici quelques exemples d'utilisation :

```
$ ./sous-chaine
Rechercher lave
 dans raton-laveur
Trouvé à la position 6
```

```
$ ./sous-chaine
Rechercher ra
 dans raton-laveur
Trouvé à la position 0
```

```
$ ./sous-chaine
Rechercher coucou
 dans raton-laveur
Absent.
```

2.1 Lecture d'un mot

On déclare un type `mot`, tableau de caractères de taille raisonnable.

```
typedef char mot[1000];
```

Si `s` est une variable de type `mot`, l'instruction `cin >> s;` permet de lire un mot tapé par l'utilisateur. La lecture s'arrête au premier espace ou à la fin de la ligne (il n'est pas possible d'entrer ainsi une chaîne de caractères contenant des espaces).

2.2 Longueur d'un mot

Après `cin >> s`, la fin du mot lu est marquée par une entrée supplémentaire qui est le caractère numéro 0. Par exemple, si l'utilisateur tape "coucou", 7 caractères sont placés au début de `s` : `'c'`, `'o'`, `'u'`, `'c'`, `'o'`, `'u'`, 0.

Écrire une fonction qui utilise ce fait pour calculer la longueur d'un mot.

```
int longueur(mot s);
```

2.3 Recherche d'un sous-mot

Écrire une fonction qui recherche un sous-mot m dans un mot s. Si m apparaît dans s à la position p, renvoyer p ; sinon, renvoyer -1.

```
int recherche(mot s, mot m);
```

Indications : commencer par calculer la longueur de s et celle de m. Ensuite, parcourir s en cherchant à chaque position si on a une occurrence de m. Si c'est le cas, utiliser l'instruction `return` à l'intérieur de la boucle.

2.4 Programme principal

Écrire un `main()` qui permette les interactions données en exemple.