

Introduction à l'algorithmique et à la programmation (Info 2)

Septième cours: Les Tableaux

Matthieu Puigt

IUT du Littoral Côte d'Opale

DUT Génie Industriel et Maintenance

`matthieu.puigt@univ-littoral.fr`

`http://www-lisic.univ-littoral.fr/~puigt/`

Retrouvez ce document sur :

`http://www-lisic.univ-littoral.fr/~puigt/teaching.html`

Année universitaire 2014–2015

Utilité des tableaux

Voir <http://www.pise.info/algo/tableaux.htm>

- On a vu les 4 instructions informatiques
- Mais il nous reste des choses à apprendre
- Imaginons que dans un programme, on ait besoin de 12 valeurs (par exemple les notes d'un étudiant sur un semestre, pour calculer sa note moyenne).
- Une façon consisterait à créer 12 variables et à les moyenner :

$$\text{Moy} \leftarrow (N1 + N2 + N3 + N4 + N5 + N6 + N7 + N8 + N9 + N10 + N11 + N12)/12;$$

- Ca marche mais c'est beaucoup plus compliqué si on doit calculer l'âge moyen des habitants d'une ville (et pire si on ne sait pas combien il y a de personnes)...
- ⇒ Les programmes informatiques permettent de rassembler toutes ces variables en une seule, au sein de laquelle chaque valeur sera désignée par un numéro

Définition

- *Un ensemble de valeurs portant le même nom de variable et repérées par un nombre, s'appelle un **tableau**, ou encore une **variable indicée***
- *Le nombre qui, au sein d'un tableau, sert à repérer chaque valeur s'appelle **l'indice***
- *Chaque élément du tableau est désigné le nom du tableau, suivi de l'indice de l'élément, entre parenthèses*

Notation et utilisation algorithmique (1)

- Retour à notre exemple
- La variable indicée s'appellera Note et nous y rentrerons les valeurs Note(0), Note(1), etc.
- **Attention : l'indice d'un tableau commence à 0 et non à 1 !!!**
(convention propre à nombreux langages — mais exceptions dont Matlab...)

Finalement :

- Un tableau est déclaré comme tel
- On définit à l'avance sa taille, en précisant son plus grand indice (convention assez répandue dans les langages informatiques)
- On définit ce qu'il va contenir (valeurs numériques, alphanumériques, booléennes)

Tableau Note(11) en Entier;

Notation et utilisation algorithmique (2)

- Gros avantage du tableau : on peut le traiter dans une boucle.

```
Tableau Note(11) en Numérique;  
Variables Moy, Som en Numérique;  
Variable i en Entier;  
Début  
Pour i  $\leftarrow$  0 à 11  
    Ecrire “Entrer la note ”, i+1 ;  
    Lire Note(i) ;  
i Suivant  
    Som  $\leftarrow$  0 ;  
    Pour i  $\leftarrow$  0 à 11  
        Som  $\leftarrow$  Som + Note(i) ;  
    i Suivant  
    Moy  $\leftarrow$  Som / 12 ;  
Fin
```

- Notez les deux boucles uniquement pour augmenter la lisibilité du programme
- L'indice peut être exprimé comme un nombre en clair, mais aussi une variable, ou une expression calculée.

Notation et utilisation algorithmique (2)

Conventions

Dans un tableau, la valeur d'un indice doit toujours :

- **être égale au moins à 0.** Donc attention, `Truc(6)` est le septième élément du tableau `Truc` !
- **être un nombre entier.** Quel que soit le langage, l'élément `Truc(3,1416)` n'existe jamais.
- **être inférieure ou égale au nombre d'éléments du tableau** (moins 1, si l'on commence la numérotation à zéro). Si le tableau `Bidule` a été déclaré comme ayant 25 éléments, la présence dans une ligne, sous une forme ou sous une autre, de `Bidule(32)` déclenchera automatiquement une erreur.

Le gag du développeur

Il consiste à confondre, dans sa tête et / ou dans un algorithme, l'indice d'un élément d'un tableau avec le contenu de cet élément. La troisième maison de la rue n'a pas forcément trois habitants, et la vingtième vingt habitants. En notation algorithmique, il n'y a aucun rapport entre `i` et `truc(i)`.

Exercices

- 1 Ecrire un algorithme qui déclare et remplit un tableau de 7 valeurs numériques en les mettant toutes à zéro.
- 2 Ecrire un algorithme qui déclare et remplit un tableau contenant les six voyelles de l'alphabet latin.
- 3 Ecrire un algorithme qui déclare un tableau de 9 notes, dont on fait ensuite saisir les valeurs par l'utilisateur.
- 4 Que produit l'algorithme suivant ?

Tableau Nb(5) en Entier;

Variable i en Entier;

Début

Pour i \leftarrow 0 à 5

 Nb(i) \leftarrow i * i;

i **Suivant**

Pour i \leftarrow 0 à 5

Ecrire Nb(i);

i **Suivant**

Fin

Peut-on simplifier cet algorithme avec le même résultat ?

5 Que produisent les algorithmes suivants ?

Tableau N(6) en Entier;
Variables i,k en Entier;

Début

$N(0) \leftarrow 1$;

Pour k $\leftarrow 1$ à 6

$N(k) \leftarrow N(k-1) + 2$;

k **Suivant**

Pour i $\leftarrow 0$ à 6

Ecrire N(i) ;

i **Suivant**

Fin

Tableau Suite(7) en Entier;
Variable i en Entier;

Début

$Suite(0) \leftarrow 1$;

$Suite(1) \leftarrow 1$;

Pour i $\leftarrow 2$ à 7

$Suite(i) \leftarrow Suite(i-1) + Suite(i-2)$;

i **Suivant**

Pour i $\leftarrow 2$ à 7

Ecrire Suite(i) ;

i **Suivant**

Fin

6 Ecrivez la fin de l'algorithme 3 afin que le calcul de la moyenne des notes soit effectué et affiché à l'écran.

Tableaux dynamiques

- On ne connaît souvent pas à l'avance la taille nécessaire pour un tableau
- On pourrait toujours déclarer un tableau avec 10000 valeurs...
- Mais on n'est pas toujours sûr que ce serait suffisant et ça prend beaucoup de place pour rien si on a besoin de moins...
- Aussi, il est possible de déclarer un tableau sans spécifier sa taille. Puis, à l'intérieur du programme, on peut spécifier sa taille grâce à l'instruction nommée **Redim**
- Voici notre exemple sur la note moyenne d'un étudiant :

```
Tableau Notes() en Entier;  
Variable Nb en Entier;  
Début  
Ecrire "Combien de notes à entrer ?";  
Lire Nb ;  
Redim Notes(Nb-1) ;  
...
```

- Il n'y a rien de dur mais il faut bien comprendre comment cela fonctionne !

Exercices

- 7 Ecrivez un algorithme permettant à l'utilisateur de saisir un nombre quelconque de valeurs, qui devront être stockées dans un tableau. L'utilisateur doit donc commencer par entrer le nombre de valeurs qu'il compte saisir. Il effectuera ensuite cette saisie. Enfin, une fois la saisie terminée, le programme affichera le nombre de valeurs négatives et le nombre de valeurs positives.
- 8 Ecrivez un algorithme calculant la somme des valeurs d'un tableau (on suppose que le tableau a été préalablement saisi).
- 9 Ecrivez un algorithme constituant un tableau, à partir de deux tableaux de même longueur préalablement saisis. Le nouveau tableau sera la somme des éléments des deux tableaux de départ. Par exemple :

Tableau 1 :

Tableau 2 :

Tableau à constituer :

1	2	3	4
5	12	-2	0
6	14	1	4

- 10 Toujours à partir de deux tableaux précédemment saisis, écrivez un algorithme qui calcule le schtroumpf des deux tableaux. Pour calculer le schtroumpf, il faut multiplier chaque élément du tableau 1 par chaque élément du tableau 2, et additionner le tout. Par exemple si l'on a :

Tableau 1 :

4	8	7	12
---	---	---	----

 et Tableau 2 :

3	6
---	---

Le Schtroumpf sera :

$$3*4 + 3*8 + 3*7 + 3*12 + 6*4 + 6*8 + 6*7 + 6*12 = 279$$

- 11 Ecrivez un algorithme qui permette la saisie d'un nombre quelconque de valeurs, sur le principe de l'exercice 7. Toutes les valeurs doivent être ensuite augmentées de 1, et le nouveau tableau sera affiché à l'écran.
- 12 Ecrivez un algorithme permettant, toujours sur le même principe, à l'utilisateur de saisir un nombre déterminé de valeurs. Le programme, une fois la saisie terminée, renvoie la plus grande valeur en précisant quelle position elle occupe dans le tableau. On prendra soin d'effectuer la saisie dans un premier temps, et la recherche de la plus grande valeur du tableau dans un second temps.
- 13 Toujours et encore sur le même principe, écrivez un algorithme permettant, à l'utilisateur de saisir les notes d'une classe. Le programme, une fois la saisie terminée, renvoie le nombre de ces notes supérieures à la moyenne de la classe.

- 14 Ecrivez un algorithme qui permette la saisie d'un polynôme d'un degré quelconque (qui sera demandé à l'utilisateur). On rappelle qu'un polynôme de degré N est de la forme :
- $$P(X) = a_0 + a_1 \cdot X + a_2 \cdot X^2 + \dots + a_N \cdot X^N$$
- 15 Ecrivez un algorithme qui permette la saisie de deux polynômes de degré quelconque (et donc pas nécessairement les mêmes) et qui en calcule leur somme. Par exemple, si les polynômes saisis sont :
- $$P_1(X) = 1 + X + X^2 \text{ et } P_2(X) = 3 + 2 \cdot X^2 - 7 \cdot X^3, \text{ le polynôme résultant sera } P_3(X) = 4 + X + 3 \cdot X^3 - 7 \cdot X^3.$$
- 16 Ecrivez un algorithme qui permette la saisie de deux polynômes de degré quelconque (et donc pas nécessairement les mêmes) et qui en calcule leur produit. Par exemple, si les polynômes saisis sont :
- $$P_1(X) = 1 + X + X^2 \text{ et } P_2(X) = 3 + 2 \cdot X^2, \text{ le polynôme résultant sera } P_4(X) = 3 + 3 \cdot X + 5 \cdot X^2 + 2 \cdot X^3 + 2 \cdot X^4.$$

Pourquoi plusieurs dimensions ?

Voir <http://www.pise.info/algo/tableauxmulti.htm>

- La raison est simple : parce que c'est plus adapté à certains problèmes !
- Prenons le cas d'un jeu de dames sur un damier de 100 cases
- Le damier peut-être modélisé par un tableau de 100 valeurs
- Chaque case du tableau modélise une case du damier, dont la valeur sera 1 si un pion est présent sur la case et 0 s'il est absent
- La première ligne du damier est représenté par les cases 0 à 9 du tableau, la deuxième ligne par les cases 10 à 19, et ainsi de suite
- Comme un pion peut se déplacer en diagonale dans les cases adjacentes de même couleur (pour simplifier), cela signifie que s'il est sur une case d'indice i , il peut aller sur les positions : $i+9$, $i+11$, $i-9$ et $i-11$
- Bien sûr, on peut programmer tout un jeu ainsi. Mais on sent que ce n'est pas simple et qu'il serait plus facile de modéliser un damier par un... damier !

Tableau à deux dimensions

- L'informatique nous permet d'utiliser des tableaux ayant **deux coordonnées**
- Cela se déclare ainsi

Tableau Cases(9,9) en Entier;

- Cela signifie : "réserve-moi un espace mémoire de 10×10 entiers et quand j'aurai besoin de l'une de ces valeurs, je le localiserai par deux indices" (comme à la bataille navale ou excel, sauf qu'on a pas une lettre et un nombre mais deux nombres)
- Dans notre problème, les choses vont être plus simples : si mon pion est situé dans la case Cases(i,j), alors il peut aller dans les positions Cases(i-1,j-1), Cases(i+1,j-1), Cases(i+1,j+1), Cases(i-1,j+1).

Remarques

- Aucune différence fondamentale entre un tableau bidimensionnel de $N \times M$ valeurs, et un tableau monodimensionnel de $N \cdot M$ valeurs.
- Se demander si le premier indice correspond aux lignes et le second aux colonnes n'a aucun sens. Dans un ordinateur, les indices d'un tableau pointent sur des adresses d'espace mémoire. Bref, choisissez votre convention et n'en changez pas.

Exercices

- 1 Ecrivez un algorithme remplissant un tableau de 6 sur 13 valeurs, avec des zéros.
- 2 Que produiront ces algorithmes ?

```
Tableau X(1,2) en Entier;  
Variables i, j, val en Entier;  
Début  
val  $\leftarrow$  1 ;  
Pour i  $\leftarrow$  0 à 1  
    Pour j  $\leftarrow$  0 à 2  
        X(i,j)  $\leftarrow$  val ;  
        val  $\leftarrow$  val + 1 ;  
    j Suivant  
i Suivant  
Pour i  $\leftarrow$  0 à 1  
    Pour j  $\leftarrow$  0 à 2  
        Ecrire X(i,j) ;  
    j Suivant  
i Suivant  
Fin
```

```
Tableau X(1,2) en Entier;  
Variables i, j, val en Entier;  
Début  
val  $\leftarrow$  1 ;  
Pour i  $\leftarrow$  0 à 1  
    Pour j  $\leftarrow$  0 à 2  
        X(i, j)  $\leftarrow$  val ;  
        val  $\leftarrow$  val + 1 ;  
    j Suivant  
i Suivant  
Pour j  $\leftarrow$  0 à 2  
    Pour i  $\leftarrow$  0 à 1  
        Ecrire X(i,j) ;  
    i Suivant  
j Suivant  
Fin
```