IUT de Lens – BUT Info S1

R1.01 – Initiation au développement

TP n^{o}3

tableaux

LES TABLEAUX À UNE DIMENSION EN JAVA

- Un tableau est un regroupement de variables de même type.
- Chaque variable du tableau est identifié par un entier qui s'appelle un indice.
- La notation `tab.length` permet d'obtenir le nombre de cases dans le tableau nommé `tab`.
- La première case du tableau porte l'indice 0, la dernière case porte l'indice `tab.length - 1`. Il est interdit d'utiliser un indice en dehors de cet intervalle.
- On crée un tableau de 10 entiers nommé *tab* en écrivant ceci : `int[] tab = new int[10];`
- Pour désigner la case d'indice `i` dans le tableau `tab`, on écrit `tab[i]`.
- `tab[i]` est une variable comme une autre. On peut donc lui affecter une valeur, ou obtenir sa valeur comme pour toutes les autres variables.

cases>	tab[0]	tab[1]	tab[2]	tab[3]
indices>	0	1	2	3

- Si on veut créer un tableau à partir d'une liste de constantes, on peut écrire `char[] voyelles = {'a', 'e', 'i' , 'o', 'u', 'y'};`
- Attention ! En Java, si t1 et t2 sont des tableaux, l'affection `t1 = t2;` ne crée pas une copie du tableau t2. On aura seulement les variables t1 et t2 qui désigneront un seul et même tableau (cf. notion de référence que vous verrez plus tard). Pour copier un tableau, il faut créer un nouveau tableau et copier chacune des cases. On ne peut pas non plus tester l'égalité de deux tableaux en écrivant simplement `t1 == t2`.

Exercice 1 : On considère le tableau `int[] tab = {4, 7, 3, 8, 2, 8};`. Écrivez un programme qui affiche les nombres de ce tableau, en les séparant par des virgules. Rien ne doit être affiché avant le premier nombre ou après le dernier.

Exercice 2 : On considère un tableau de N entiers (N étant une constante). Initialisez le contenu de ce tableau avec les nombres saisis par l'utilisateur.

Exercice 3 : Soit un tableau de N entiers (N étant une constante) saisis par l'utilisateur. Calculez et affichez la moyenne des nombres de ce tableau.

Exercice 4 : Soit un tableau de N entiers (N étant une constante) saisis par l'utilisateur. Inversez l'ordre des éléments dans le tableau (le premier doit devenir le dernier, le dernier doit devenir le premier, etc.)

Exercice 5 : Soit un tableau de N entiers (N étant une constante). Écrivez un programme qui compte le nombre d'occurrences (apparitions) d'un entier e.

Exercice 6 : Soit un tableau de N entiers (N étant une constante). Écrivez un programme qui trouve la position du plus petit élément dans l'intervalle d'indices $[beg..end[$.

Exercice 7 : Soit un tableau de N entiers (N étant une constante). En utilisant le code de l'exercice précédent, triez le tableau par ordre croissant avec un tri par sélection.

Exercice 8 : Soit un tableau de N entiers (N étant une constante). On suppose qu'il n'y a que les t premières cases qui sont utilisées (les dernières cases ne contiennent pas d'information pertinente). On suppose aussi que ces t premiers entiers sont triés par ordre décroissant. Écrivez un programme qui insère un entier e à la bonne position dans le tableau. À la suite de l'insertion, le tableau contient $t + 1$ entiers qui sont toujours triés par ordre décroissant.

Exercice 9 : Soit un tableau de N entiers (N étant une constante). En utilisant le code de l'exercice précédent, triez le tableau par ordre décroissant avec un tri par insertion.

Exercice 10 : Soit un tableau de N entiers (N étant une constante). Triez le tableau par ordre décroissant en utilisant un tri à bulles.

Exercice 11 : Soit un tableau de $2N$ entiers (N étant une constante). Ce tableau contient des paires d'entiers. Pour la première paire, le premier entier se trouve à l'indice 0 et le second à l'indice 1. Pour la deuxième paire, le premier entier se trouve à l'indice 2 et le second à l'indice 3. Écrivez un programme qui trie les paires du tableau par ordre croissant dans l'ordre lexicographique sur les paires (i.e. l'ordre du dictionnaire). Une paire (a, b) est strictement plus petite qu'une paire (a', b') ssi $a < a'$ ou ($a = a'$ et $b < b'$).

Exercice 12 : Soit un tableau t de N entiers (N étant une constante). On suppose que ce tableau peut contenir n'importe quel entier, mais que le nombre d'entiers différents ne peut pas dépasser M. Créez un tableau de paires d'entiers (stockés dans 2 cases consécutives du tableau) tels que le premier élément de la paire est une valeur du tableau t et le second élément de la paire est le nombre d'occurrences de cette valeur dans le tableau t. À la fin, vous devez afficher le nombre d'occurrences par ordre croissant des valeurs.

Exercice 13 : Soient un entier n, une base b et un tableau t de N entiers (N étant une constante). Vous devez ranger dans la case $t[i]$ le chiffre d'indice i du nombre n écrit en base b.

Exercice 14 : Soient un entier n, une base b et un tableau t de N entiers (N étant une constante). On considère que t contient les chiffres d'un nombre écrit en base b ($t[0]$ étant le chiffre de plus faible poids). Vous devez ranger dans n la valeur du nombre (en base 10).

Exercice 15 : Soient une base b et trois tableaux x, y et r de N entiers (N étant une constante). Ces tableaux contiennent les chiffres de nombres écrits en base b. Vous devez additionner les nombres x et y et ranger le résultat dans r.

Exercice 16 : Soient une base b et trois tableaux x, y et r de N entiers (N étant une constante). Ces tableaux contiennent les chiffres de nombres écrits en base b. Vous devez multiplier les nombres x et y et ranger le résultat dans r.

Polynômes

L'objectif de ce sujet est d'effectuer des calculs sur des polynômes à une variable. Un polynôme $P(x)$ est une expression de la forme $a_0 + a_1.x + a_2.x^2 + a_3.x^3 + ...a_n.x^n$

On veut manipuler des polynômes de degré n au maximum ce qui signifie qu'ils ne contiennent pas de terme x^i avec $i > n$ (on pourra prendre $n = 5$). On choisit donc de conserver les coefficients des x^i du polynôme $P(x)$ dans un tableau t tel que

$$P(x) = \sum_{i=0}^{n} t[i].x^i$$

Par exemple, le polynôme $3 + x^2 - 5x^3$ sera représenté par le tableau

3	0	1	-5	0	0

Exercice 17 : Écrivez le code pour lire et afficher des polynômes.

Exercice 18 : Écrivez ensuite l'addition de 2 polynômes.

Exercice 19 : Écrivez le code pour calculer la valeur $P(x)$ que prend un polynôme P en un point x.

Exercice 20 : Écrivez le code pour stocker dans un polynôme Q la dérivée d'un polynôme P. On rappelle que la dérivée de $a_i.x^i$ est $i.a_i.x^{i-1}$ et il suffit d'appliquer cette formule à chaque monôme.

Exercice 21 : Écrivez le code pour stocker dans un polynôme Q la primitive d'un polynôme P. Si cette primitive a un degré supérieur à n, vous devez afficher un message d'erreur. On rappelle que la primitive de $a_i.x^i$ est $\frac{a_i}{i+1}.x^{i+1}$ et il suffit d'appliquer cette formule à chaque monôme. Dans la primitive que vous calculerez, le coefficient de x^0 (la constante a_0) sera égal à 0.

Exercice 22 : Écrivez le code qui calcule l'intégrale entre a et b d'un polynôme P. Pour obtenir cette valeur, il suffit de calculer la valeur de la primitive de P pour $x = b$ et de lui soustraire la valeur de la primitive de P pour $x = a$. Cette valaur donne l'aire de la surface comprise entre la courbe et les 3 droites d'équation $y = 0$ (l'axe des abcisses), $x = a$ et $x = b$.

Exercice 23 : Écrivez la multiplication de 2 polynômes. Si le polynôme résultat a un degré supérieur à n, vous devez afficher un message d'erreur.