

TD 2 : Tableaux

Exercices table : 3, 4, 7, 8, 9, 10, 11, 12, 13, 14, 18, 19.

Rappel : respectez les consignes de la page *Fonctionnement*.

1. Placez-vous dans le répertoire adéquat demandé dans la page *Environnement*.
2. **Familiarisation** Tapez et exécutez le programme *Ex3.java* de la feuille d'exemples du cours, en respectant les consignes de la page *Fonctionnement*.
3. Création d'un tableau de 5 cases sans initialiseur. Affichez la case 0, la 1 et la 4. Affichez la taille du tableau. Remplissez ce tableau (sans boucle), avec au moins 4 valeurs non nulles. Affichez à nouveau la case 0, la 1 et la 4.
4. Création d'un tableau de 5 cases *avec* initialiseur, avec les mêmes valeurs que l'exercice précédent. Affichez la case 0, la 1 et la 4. Affichez la taille du tableau.
5. Création d'un tableau de 8 cases par initialiseur, avec des valeurs strictement négatives, strictement positives, et des valeurs nulles. Affichez toutes les cases du tableau, sans boucle.
6. Créez un tableau par initialiseur, affichez son adresse, et expliquez ce qu'on voit apparaître. On se restreindra à des tableaux de taille 5, ainsi que dans toute la suite du TD.
7. Affichage des éléments strictement positifs d'un tableau de 5 cases créé par initialiseur. On affichera évidemment le tableau pour tester, ainsi que dans toute la suite.
8. Affichez le 3ème élément de la ligne de commande, puis le 1er, puis le 2ème. Testez en passant 5 paramètres sur la ligne de commande, des entiers et des chaînes.
9. Affichez le nombre de paramètres de la ligne de commande.
10. Affichez tous les paramètres de la ligne de commande.
11. On suppose que les paramètres de la ligne de commande sont des entiers. Affichez-les tous mais incrémentés chacun de 1.
12. Saisie d'un tableau d'entiers passés sur la ligne de commande (ne pas rentrer la taille). Dans toute la suite du TD, les tableaux seront lus ainsi sur la ligne de commandes.
13. On suppose que sont passées sur la ligne de commande des étudiants avec leur nombre de cours et leur année de naissance comme suit. Attention à l'ordre : d'abord année de naissance du premier étudiant, puis son nom, puis son nombre de cours, puis année de naissance d'un autre étudiant, puis son nom, puis son nombre de cours, etc. On considère que l'on est en 2009, et que quelqu'un né en 2000 a 9 ans pour simplifier. Stockez toutes ces informations dans des tableaux et/ou des variables : tableau des années, tableau des noms, etc. Affichez toutes ces personnes, une par une, chacune avec ses informations dans l'ordre suivant (attention au nouvel ordre) : nom, âge (attention : pas l'année de naissance), nombre de cours.
14. Afficher n lignes d'étoiles (*) selon le contenu d'un tableau de n entiers positifs ou nuls. Ex : 3, 0, 5 donne 3 étoiles, ligne vide, 5 étoiles.
15. Mettre à zéro tous les éléments d'un tableau existant (*new* interdit).
16. Modifier un tableau, en changeant tous ses éléments strictement positifs en leur opposé.
17. Faites une copie d'un tableau existant, mais la copie sera dans l'ordre inverse.

18. Trouver le minimum d'un tableau.
19. Décaler de une case vers la droite tous les éléments d'un tableau, le dernier sera mis dans la première case (*new* interdit).
20. Trouver la case d'un tableau contenant le plus petit élément, décaler toutes les cases à gauche de cette case d'une case vers la droite, mettre ce plus petit élément dans la première case (*new* interdit). On a le même contenu du tableau, mais le minimum est dans la première case.
21. Reprendre l'exercice 20, recommencer avec le 2ème plus petit élément, puis le 3ème plus petit, etc. (*new* interdit). A la fin, le tableau est trié: c'est le *tri par insertion en place*.
22. *Déroulement mémoire. Travail personnel.*
On considère l'exécution du programme suivant (il compile sans erreur).
 - (a) Donner la configuration de la mémoire immédiatement après l'exécution de la ligne contenant le commentaire: `// ici`. Vous utiliserez un croquis détaillé et l'algorithme vu en cours et TD. En particulier, on allouera obligatoirement les cases mémoire dans l'ordre croissant à partir de 101. Comme en cours et en TD, si une case contient successivement plusieurs valeurs, on les écrira de gauche à droite dans la case en les barrant d'un seul trait au fur et à mesure. De même, on ne réutilisera pas les zones de fonctions.
 - (b) Donner le dernier état de la mémoire avant l'arrêt du programme.
 - (c) Dire ce qu'affiche ce programme.

```

class MemoireTableau {
    public static void main (String[] args ) {
        int[] t = new int[3];
        t[0] = t[2];
        t[2] = 1;
        int n = t[1];
        t[n]++;
        int m = t[n];
        n = t[t[n]];
        System.out.println(m+" "+n); // ici
    }
}

```

Si vous avez fini tous les exercices ci-dessus avant la fin du TD

Choisissez ce que vous voulez dans la suite. Ces exercices ne sont pas très durs, et ne contiennent pas de concept nouveau par rapport à ceux ci-dessus. Ils sont là simplement à titre d'entraînement de pratique si nécessaire.

23. On suppose que sont passées sur la ligne de commande des personnes avec leur âge comme suit. Nom d'une personne puis son âge, nom d'une autre puis son âge, etc. On suppose que c'est leur anniversaire aujourd'hui. Affichez toutes ces personnes, chacune avec son nouvel âge.
24. Somme éléments en position impaire dans tableau (1, 3, 5, etc.). Saisie sur la ligne de commande comme ci-dessus, affichage tableau (pour vérifier).
25. Minimum d'un tableau.
26. Décaler de une case vers la gauche les éléments d'un tableau existant (*new* interdit), et le premier sera mis dans la dernière case. Saisie, affichage, décalage, affichage.
27. Création d'un nouveau tableau à partir d'un tableau existant. Il contiendra les éléments de l'ancien, décalés de une case vers la gauche, et le premier sera dans la dernière case. Saisie, affichage, création, affichage des deux tableaux.