

Tableaux de tableaux (TD 7)

Exercices table: 3, 4, 5, 6, 7 8.

Rappel: Respectez les consignes de la page *Fonctionnement* et celles données dans les feuilles précédentes.

1. Placez-vous dans le répertoire adéquat demandé dans la page *Environnement*.
2. **Familiarisation** Tapez et exécutez le programme suivant de la feuille d'exemples du cours, en respectant les consignes de la page *Fonctionnement*: *Ex4bis.java*.
3. Création par initialiseur d'un tableau irrégulier de tableaux de 3 lignes, avec une de 4 éléments, une de 0 et une de 5, et affichage. Sans fonction. Dans tout ce TD, il s'agit de tableaux de tableaux d'entiers.
4. Même exercice, mais avec fonctions.
5. Somme des éléments d'un tableau irrégulier de tableaux, sans fonction.
6. Trouver le maximum d'un tableau irrégulier de tableaux, avec fonctions.
7. Afficher nombre de lignes d'un tableau irrégulier à deux indices, sans fonction.
8. Afficher nombre d'éléments d'un tableau irrégulier à deux indices, avec fonctions.
9. Afficher un tableau régulier carré, mais avec toutes ses lignes à l'envers, sans fonction.
10. Afficher la diagonale montante d'un tableau régulier carré, sur une ligne, sans fonction.
11. Addition, puis multiplication de matrices non carrées. Le résultat sera mis dans une nouvelle matrice. Sans fonction.
12. Saisie d'un tableau à deux indices irrégulier d'entiers sur la ligne de commande. Vous entrerez le nombre de lignes et la taille de chaque ligne. Avec fonctions.

13. *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 MemoireTab2Tab {
    static int c;
    static int f (int[] s, int n) {
        n++;
        c--;
        s[n] = 2;
        int[] t = s;
        t[2] = 3;
        return n;
    }
    public static void main (String[] args ) {
        int[] t = new int[3];
        t[2] = 1;
        int n = t[1];
        t[n]++;
        int m = t[n];
        n = t[t[n]];
        System.out.println(m+" "+n);
        int[] [] u = new int[2] [];
        u[n] = new int[m+1];
        u[m] = t;
        u[n][u[0].length-1]++;
        System.out.println(m+" "+n+" "+c); // ici
        int[] [] v = new int[2] [];
        v[0] = t;
        m = f(v[0], n);
        System.out.println(m+" "+n+" "+c);
    }
}
```