

-----

Le TD est consacré à des révisions du langage C, via la description de types abstraits de données (T.A.D.). L'objectif est à nouveau de préparer des descriptions analogues en langage C++, qui seront développées dans le TP suivant.

### Exercice Le T.A.D. *Matrice*

Analyser en détail l'interface fonctionnelle de l'ensemble des matrices carrées d'ordre N, à coefficients entiers, (T.A.D. *Matrice* / fichier "*Matrice.h*" en annexe).

Justifier l'usage :

- d'un tableau à une seule dimension (langage C) pour mémoriser tous les éléments d'une instance,
- du type référence pour les paramètres formels qui désignent une matrice,
- du spécificateur **const** pour chacun de ces paramètres.

La description inclut les trois opérations de base (+, - et \*), y compris la multiplication externe (multiplication par un scalaire). Ajouter à la description la déclaration de l'opération unaire de changement de signe.

Une fonction **toString**, analogue à celle développée pour le T.A.D. "Fraction" (Feuille N° 1), permettra le transcodage global de chaque matrice sous forme d'une chaîne de caractères formatée. Cette fonction facilite la description des modules de tests unitaires. Pour ce transcodage, les coefficients de la matrice seront affichés sur trois positions (cadrage individuel à droite) et séparés par une tabulation. Un saut de ligne unique marquera la fin de chaque vecteur ligne de la matrice cible.

La description intègre les accesseurs **element** (élément quelconque de position (i, j),  $1 \leq i \leq N$ ,  $1 \leq j \leq N$ ), **ligne** (ligne de rang i,  $1 \leq i \leq N$ ) et **colonne** (colonne de rang j,  $1 \leq j \leq N$ ).

Une fonction **remplir** permettra de copier dans la ligne de rang i d'une matrice quelconque ( $1 \leq i \leq N$ ), les éléments d'un tableau, fourni en paramètre.

Détailler le code source en langage C du T.A.D. *Matrice*, conforme à l'interface fonctionnelle fournie.

Exécuter avec succès tous les modules de tests unitaires fournis en annexe.

### Exercice Compléments pour le T.A.D. *Matrice*

Compléter la description obtenue en fin de l'exercice par la prise en compte des nouvelles exigences fonctionnelles suivantes :

- calcul du déterminant d'une matrice NxN,
- inversion d'une matrice NxN,

Détailler le code source en langage C des compléments correspondants.

Développer et exécuter avec succès tous les modules de tests unitaires nécessaires pour valider ces compléments.