

Développement initiatique

Sujet 4 : tableaux à deux dimensions

Notions à acquérir

- Définition et utilisation des tableaux à deux dimensions.
Exemple : on déclare un tableau d'entiers à deux dimensions (10 lignes et 20 colonnes) de la manière suivante : `mat : tableau de (10 tableaux de 20 entiers)` ou bien :
`mat : matrice de 10 lignes et 20 colonnes d'entiers`
- Parcours total et parcours partiel.

1 Opérations sur matrices d'entiers

Le but de l'exercice est d'écrire des fonctions implantant les opérations ci-dessous sur des matrices d'entiers, et de les appeler dans une fonction de test.

1. Addition de deux matrices données (de mêmes dimensions).
2. Somme des éléments des deux diagonales d'une matrice carrée d'entiers donnée. Signature :
`fonction sommeDiagos(mat : tableau de tableaux d'entiers) retourne entier`
3. Détermination de valeurs nulles :
 - (a) compter le nombre d'éléments de valeur nulle dans une matrice donnée ;
 - (b) savoir s'il existe au moins une valeur non nulle dans une matrice donnée ;
 - (c) savoir sur quelle ligne d'une matrice donnée on trouve le plus de valeurs nulles ;
 - (d) savoir si, dans une matrice donnée, il existe une ligne qui comporte plusieurs valeurs nulles.
4. (optionnel) Calcul du produit de deux matrices données (dont les dimensions respectives permettent d'effectuer ce produit).

2 Carré magique

Ecrire une fonction qui permet de créer un carré magique parfait ayant un nombre impair de lignes et de colonnes. La fonction renvoie la matrice correspondante.

Pour cela, on initialise la matrice avec des zéros, puis on positionne la valeur 1 sur la première ligne, colonne du milieu.

Lorsque cela est possible, les nombres suivants sont positionnés un après l'autre, sur la ligne précédente et la colonne suivante du dernier nombre inscrit.

Si la ligne précédente n'existe pas, on se positionne sur la dernière ligne. Lorsque la colonne suivante n'existe pas on revient à la première colonne¹.

Par ailleurs, si la case déterminée contient déjà un nombre, on revient se positionner sur la case en dessous de la dernière case inscrite (on admet que celle-ci est toujours libre).

Exemple avec 3 lignes et 3 colonnes (utilisez la stratégie ci-dessus pour placer les deux derniers nombres) :

	1	6
3	5	7
4		2

3 Des matrices carrées avec un motif particulier

Ecrire des fonctions prenant une dimension entière positive n en paramètre et retournant une matrice carrée $n \times n$ qui rangent les n^2 premiers entiers de la manière suivante (pour des matrices 4×4) :

1. Matrice « rangée » :

```
1  2  3  4
5  6  7  8
9  10 11 12
13 14 15 16
```

2. Matrice « serpent » :

```
1  2  3  4
8  7  6  5
9  10 11 12
16 15 14 13
```

3. Matrice « diagonale » :

```
1  3  6  10
2  5  9  13
4  8  12 15
7  11 14 16
```

4. Matrice « spirale » :

```
1  2  3  4
12 13 14 5
11 16 15 6
10 9  8  7
```

4 Sudoku (optionnel)

Vérifier qu'une grille de Sudoku $n \times n$ entièrement remplie est valide. Un Sudoku $n \times n$ (ou de taille n) comporte n^2 lignes de n^2 cases, n^2 colonnes de n^2 cases et n^2 carrés de n^2 cases.

1. Pour utiliser une fonction de modulo en Java, vous pouvez faire appel à `Ut.modulo2(a,b)` qui renvoie un nombre entre 0 et $b - 1$ si b est positif.