

## TD9 : éléments de correction

### Exercice 2

Procédure : **EcrireCarreMagique**

Précondition : dimensions strictement supérieures à 0

Entrée :  
- carreMagique : **tableau 2D d'entiers**  
- chemin : **chaîne de caractères**

Local :  
- f : **flux d'écriture**

Sortie :  
**void**

Début

```
Soit f = flux d'écriture initialisé sur chemin
// écrire les dimensions du tableau sur la première ligne
Ecrire(f, "{Longueur(carreMagique, 0)},{Longueur(carreMagique, 1)}\n") // ex : 3,3
// écrire le tableau ligne par ligne (séparateur '\n'), colonne par colonne (séparateur ',')
Pour ligne de 0 à Longueur(carreMagique, 0) - 2 avec un pas de +1 Faire
    Pour colonne de 0 à Longueur(carreMagique, 1) - 1 avec un pas de +1 Faire
        Soit valeur = Chaîne(carreMagique[ligne, colonne]) // conversion en chaîne
        Écrire("{valeur},")
    Fin Pour
    Soit valeur = Chaîne(carreMagique[ligne, Longueur(carreMagique, 1) - 1])
    Écrire("{valeur}\n")
Fin Pour
FermerFlux(f)
```

Fin

Fonction : **LireCarreMagique**

Entrée :  
- chemin : **chaîne de caractères**

Local :  
- f : **flux de lecture**

Sortie :  
- carreMagique : **tableau 2D d'entiers**

Début

```
Soit f = flux de lecture initialisé sur chemin
// lire les dimensions du tableau sur la première ligne
Soit ligne = LireLigne(f)
Soit details = Séparer(ligne, ',') // ex : tableau ["3", "3"]
Soit lignes = Entier(details[0])
Soit colonnes = Entier(details[1])
// création du tableau
Soit carreMagique = tableau 2D de lignes*colonnes entiers
// lecture du tableau ligne par ligne (séparateur '\n'), colonne par colonne (séparateur ',')
Soit i = 0
Faire
    Soit ligne = LireLigne(f)
    Soit details = Séparer(ligne, ',')
    Pour j de 0 à colonnes avec un pas de +1 Faire
        Soit carreMagique[i, j] = Entier(details[j])
    Fin Pour
    Soit i = i + 1
Tant Que ligne n'est pas nulle
FermerFlux(f)
Retourner carreMagique
```

Fin