
L1 MIASHS - ALGORITHMIQUE

FICHE EXERCICES 8

LES TABLEAUX 2D

EXO 1

Créez un algorithme qui prend en ENTREES un nombre de lignes **nl**, un nombre de colonnes **nc** et un tableau 2D de nl lignes et nc colonnes de NUMERIQUE. L'algorithme donne en SORTIES le maximum du tableau.

EXO 2

Créez un algorithme qui prend en ENTREES un nombre de lignes **nl**, un nombre de colonnes **nc** et un tableau 2D de nl lignes et nc colonnes de NUMERIQUE. L'algorithme donne en SORTIES la somme maximale sur toutes les lignes du tableau. Par exemple, pour le tableau ci-dessous, la somme maximale est 24 (deuxième ligne) :

8	4	5	6
2	5	8	9
7	2	1	3

EXO 3

Créez un algorithme qui prend en ENTREES un nombre de lignes **nl**, un nombre de colonnes **nc** et un tableau 2D de nl lignes et nc colonnes de numériques. L'algorithme donne en SORTIES la somme maximale sur toutes les colonnes du tableau. Par exemple, pour le tableau ci-dessous, la somme maximale est 18 (dernière colonne) :

8	4	5	6
2	5	8	9
7	2	1	3

EXO 4

Créez un algorithme qui prend en ENTREES un nombre **n** et un tableau 2D de n lignes et n colonnes de numériques. L'algorithme donne en SORTIES un boolean qui indique si la tableau est symétrique ou non. Un tableau est symétrique si pour tout i, j : $t[i,j] = t[j,i]$

EXO 5

Soit un plateau de TicTacToe représenté par un tableau 2D de 3 lignes et 3 colonnes de CHAINES. Chaque case contient " " ou "X" ou "O". Créez un algorithme prenant en ENTREE un tel tableau et indiquant en SORTIE si il contient une solution gagnante. On utilise pour cela une CHAINE qui indique " " si la configuration n'est pas gagnante, "X" si le joueur X est gagnant ou "O" si le joueur O est gagnant.

EXO 6

Soit un tableau 2D **termes** contenant des NUMERIQUE. Soit un tableau 1D **chemin** contenant des CHAINE de valeurs « GAUCHE », « DROITE », « HAUT » et « BAS ». On part de la case de coordonnées (1,1) de **termes**. On parcourt alors le tableau **chemin** pour connaître la case suivante à visiter dans **termes**. On fait la somme de toutes les cases jusqu'à avoir terminer de parcourir **chemin**.

Créez un algorithme prenant en ENTREE ces deux tableau (n'oubliez pas leurs taille) et donnant en SORTIE la somme des cases visitées. On supposera que le chemin ne sort pas du tableau 2D.

Bonus : si le chemin sort du tableau 2D, alors on « entre » par l'autre côté (par exemple si on est en bout de ligne et qu'on va à DROITE, on revient en début de ligne)