

Durée : 1h

Toutes les fonctions que vous écrirez doivent être clairement commentées avec des affirmations significatives (antécédents, conséquents, invariants). Vous prendrez soin de définir les bons paramètres et les bons types des données manipulées. Pensez à définir des fonctions auxiliaires si cela est nécessaire.

Soit une matrice d'entiers $n \times n$, écrivez la fonction **somme** qui renvoie la somme des valeurs des éléments qui se trouvent sur les deux diagonales. Attention aux indices !

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

[illegible]

Exercice 2

Une matrice $m \times n$ dont les dimensions sont lues sur l'entrée standard contient des entiers quelconques, à l'exception de la case $(0, 0)$ qui contient la valeur 0.

On veut déplacer un jeton sur la matrice de haut en bas et gauche à droite au gré de la valeur d'un dé. On lance répétitivement le dé qui renvoie à chaque fois une valeur comprise entre 1 et 6 correspondant au nombre de cases à avancer. Les tirages du dé s'arrêtent dès qu'on dépasse la dernière case.

Écrivez la procédure `somme` qui prend une matrice `mat` et ses dimensions en paramètres et qui, en partant de la case $(0, 0)$, renvoie la somme des valeurs des cases sur lesquelles le jeton passe. La fonction `lancerDé()` est donnée et renvoie de façon aléatoire un entier compris entre 1 et 6.

Par exemple, pour la matrice 4×3 donnée ci-dessous :

$$\begin{vmatrix} 0 & 1 & 2 \\ 3 & 4 & 5 \\ 6 & 7 & 8 \\ 9 & 10 & 11 \end{vmatrix}$$

et avec la suite de tirages suivante 2, 4, 3, 1, 4, la procédure `somme` renverra 27 (*i.e.* $27 = 2 + 6 + 9 + 10$). Notez que le dernier tirage 4 fait sortir de matrice.

[illegible]

RLE (*Run Length Encoding*) est une technique de compression de données, anciennement utilisée pour la compression d'images. Le principe est rudimentaire et très simple : une séquence de caractères c identiques est *remplacée* par :

où **marqueur** est un caractère spécial, si possible, peu fréquent dans la suite de caractères à comprimer et L la longueur de la séquence de caractères c codée sur 1 seul caractère. Si L est codée sur un caractère, cela veut dire que ce codage ne comprime qu'une suite d'au plus 9 caractères identiques. Si la suite fait plus de 9 caractères, les 9 premiers seront comprimés, puis la suite de la séquence sera considérée comme une nouvelle séquence à coder. Lorsque le marqueur apparaît dans les données à comprimer, il est remplacé par « **marqueur 0** ».

Écrivez la procédure `decompresser` qui lit toute l'entrée standard (avec `getchar` uniquement) une suite de caractères comprimée selon la méthode *RLE* avec `'#'` comme marqueur et qui écrit sur la sortie standard (avec `putchar` uniquement) sa forme décomprimée. La marque est passée en paramètre de la procédure.

D'autre part, vous penserez à signaler une erreur si l'entrée standard n'est pas conforme à la syntaxe *RLE*.

[illegible]

This image shows a single sheet of white paper with horizontal ruling lines. The lines are evenly spaced and run across the width of the page. There are no margins, text, or other markings on the paper.