

Exercice 1 : Parcourir des tableaux à deux dimensions

On a défini un type tableau par :

```
const N = 5 ; // modifiable
type T_INDICE = 1..N ;
type T_ELEMENT = 1..N*N ;
type T_TABLEAU is array[T_INDICE,T_INDICE] of T_ELEMENT ;
```

Nous allons écrire des procédures pour parcourir de différentes manières un tel **T_TABLEAU**.

Q 1 . Pour éventuellement tester nos procédures, on va d'abord remplir ce tableau. Écrivez une procédure **remplir** à un paramètre **t** de type **T_TABLEAU** qui remplit ce tableau ligne par ligne avec les éléments de **T_ELEMENT**.

Après l'exécution de cette procédure, le tableau **t** doit avoir pour valeur (si **N** vaut 5)

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20
21	22	23	24	25

Nous avons vu en cours les algorithmes pour parcourir ce tableau **t** suivant les manières suivantes :

- ligne par ligne
- colonne par colonne

Voici d'autres manières curieuses de parcourir ce tableau

Q 2 . Écrivez une procédure **afficherBoustro** à un paramètre **t** de type **T_TABLEAU** qui affiche le contenu des cases de ce tableau, lors d'un parcours en *boustrophédon* (du grec : bous : bœuf , strephein : tourner), c'est-à-dire par ligne mais en zig-zag, donc la première ligne de gauche à droite, la deuxième de droite à gauche, la troisième de gauche à droite, etc ... comme le fait le bœuf tirant la charrue.

Si **N** vaut 5 et **t** a été rempli par la procédure **remplir** on doit avoir

1 2 3 4 5 10 9 8 7 6 11 12 13 14 15 20 ...

Q 3 . On peut aussi parcourir ce tableau en colimaçon en partant de la case en haut à gauche.

Écrivez une procédure **afficherColi** à un paramètre **t** de type **T_TABLEAU** qui affiche le contenu des cases de ce tableau, lors d'un parcours en colimaçon.

Si **N** vaut 5 et **t** a été rempli par la procédure **remplir** on doit avoir

1 2 3 4 5 10 15 20 25 24 23 22 21 16 11 6 7 8 9 14 19 18 17 12 13

Q 4 . On peut aussi parcourir ce tableau en diagonale et on a deux possibilités ↗ ou ↘.

Dans le premier cas on aurait

1 6 2 11 7 3 16 12 8 4 21 17 13 9 5 22 18 14 10 23 19 15 24 20 25

Dans le second

5 4 10 3 9 15 2 8 14 20 1 7 13 19 25 6 12 18 24 11 17 23 16 22 21

Écrivez les procédures **afficher_SO_NE** et **afficher_NO_SE**

Q 5 . Enfin on peut tout bêtement parcourir ce tableau au hasard mais sans visiter deux fois le même case.

Écrivez une procédure **afficherHasard**

Exercice 2 : Le code de Vigenère

En 1585, Blaise de Vigenère a exposé une méthode pour coder un message, et il faudra plusieurs siècles pour en venir à bout.

Sa méthode utilise un carré de lettres et une clef secrète.

Le carré de lettres est obtenu en écrivant l'alphabet 26 fois en faisant à chaque fois une permutation circulaire d'une lettre

A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A
C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C
E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D
F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E
G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F
H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G
I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H
J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I
K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J
L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K
M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L
N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M
O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N
P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
Q	R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
R	S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
S	T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
T	U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
U	V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
V	W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U
W	X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V
X	Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W
Y	Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X
Z	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y

Si le message clair est “nous attaquerons à l’aube” et que la clef est “henri”, on commence par écrire, l’un sous l’autre, le message clair et la clef, répétée autant de fois que nécessaire. Le message codé est alors obtenu en regardant la lettre qui apparaît à l’intersection de la colonne de la première lettre du message, N, et de la ligne de la première lettre de la clef, H, dans le carré de Vigenère: c’est un U. On regarde ensuite la lettre à l’intersection de la colonne de la deuxième lettre du message clair, O, et de la ligne de la deuxième lettre de la clef E (c’est un S), etc.

n	o	u	s	a	t	t	a	q	u	e	r	o	n	s	a	l	a	u	b	e
H	E	N	R	I	H	E	N	R	I	H	E	N	R	I	H	E	N	R	I	H
U	S																			

- Q 1 .** Terminez le codage du message.
- Q 2 .** Que proposez-vous comme structure de données pour stocker le “carré de Vigenère” ?
- Q 3 .** Écrivez une fonction `codage` à deux paramètres `l`, la lettre à coder, et `c`, la clef du codage, qui renvoie le codage de la lettre `l`.
- Q 4 .** Écrivez une fonction `codage` à deux paramètres `m`, le message à coder, et `c`, la clef du codage, qui renvoie le message codé.
- Q 5 .** Écrivez les fonctions de décodage

(plus d’info : <http://www.lifl.fr/wegrzyno/enigma/enigma/vigenere.html>)