



## Problème I - Parité des entiers naturels

Ecrivez un programme qui détermine si un nombre entier naturel lu du clavier est pair, pairplus, super-pairplus ou ultra-pairplus.

### Définitions

Un entier naturel est appelé **pair** s'il est divisible par 2 (autrement dit, si le reste de la division entière du nombre par 2 est 0).

Un entier naturel est appelé **pairplus** s'il est pair et en plus composé que de chiffres pairs.

Un entier naturel est appelé **super-pairplus** s'il est pairplus et si en plus ses diviseurs, à part de 1, ne sont que des entiers naturels pairs.

Un entier naturel est appelé **ultra-pairplus** s'il est pairplus et si en plus ses diviseurs, à part de 1, ne sont que des entiers naturels pairplus.

### Exemples

14 et 92 sont des entiers pairs.

82 et 408 sont des entiers pairplus.

64 et 2048 sont des entiers super-pairplus.

4 et 8 sont des entiers ultra-pairplus.

### Entrées et sorties du programme

⇒ Le programme lit du clavier un nombre entier naturel quelconque.

⇐ Le programme affiche à l'écran un ou plusieurs des textes suivants:

"Entier naturel impair"	si le nombre lu du clavier est un entier naturel, mais n'est pas pair;
"Entier naturel pair"	si le nombre lu du clavier est pair;
"Entier naturel pairplus"	si le nombre lu du clavier est pairplus;
"Entier naturel super-pairplus"	si le nombre lu du clavier est super-pairplus;
"Entier naturel ultra-pairplus"	si le nombre lu du clavier est ultra-pairplus;

ou encore un message d'erreur de votre choix si le nombre lu ne correspond pas à un entier naturel.

### Restrictions

Les valeurs lues du clavier sont toujours numériques et se situent dans l'intervalle [-20.000; 20.000].



Sauvegardez le programme dans le répertoire **N On Server (H:)** où N désigne votre nom d'utilisateur sous le nom **PARITE.xxx** (où xxx=PAS ou XXX=C). Le même répertoire doit également contenir le fichier binaire exécutable **PARITE.EXE** correspondant au programme.

## Problème I I - Enigma

Ecrivez un programme qui trouve un mot dans un tableau carré  $N \times N$  ( $N$  étant un entier naturel) composé des 26 lettres majuscules de l'alphabet. Le mot peut être inscrit horizontalement de la gauche vers la droite ou inversement, verticalement du haut en bas ou inversement ou encore diagonalement du haut en bas ou inversement.

### Exemple: tableau 8´8

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

Ce tableau contient les mots "INTEL" et "PENTIUM".

### Entrées et sorties du programme

- ⇒ L'entrée du programme est le fichier texte **ENIGMA.IN** où la première ligne contient l'entier naturel  $N$ , la deuxième la longueur du mot à trouver, la troisième le mot à trouver et les lignes suivantes les lignes du tableau carré.
- ⇐ La sortie du programme est le fichier texte **ENIGMA.OUT** qui contient un des deux textes OUI (si le mot est contenu dans le tableau) ou NON (dans le cas contraire).

### Exemples

<b>ENIGMA.IN</b>	<b>ENIGMA.OUT</b>		<b>ENIGMA.IN</b>	<b>ENIGMA.OUT</b>
<pre>8 5 I N T E L A T Z L U L D W F O D W E R A S E R C T I M L Q C E N L G J K K X I E Q P N D S S A I R K O N D M U I T N E P F Y O K S H B P G</pre>	<pre>O U I</pre>		<pre>8 7 C E L E R O N A T Z L U L D W F O D W E R A S E R C T I M L Q C E N L G J K K X I E Q P N D S S A I R K O N D M U I T N E P F Y O K S H B P G</pre>	<pre>N O N</pre>

### Restrictions

La valeur pour  $N$  se situe dans l'intervalle entier  $[2; 20]$ . La longueur des mots à trouver se situe dans l'intervalle  $[2; N]$ .

### Jeu de tests

Afin de tester votre programme, les deux tableaux d'entrée ci-dessus sont sauvegardés sous les noms **ENIGMA1.IN** et **ENIGMA2.IN** dans le répertoire .....



Sauvegardez le programme dans le répertoire **N On Server (H:)** où  $N$  désigne votre nom d'utilisateur sous le nom **ENIGMA.xxx** (où xxx=PAS ou XXX=C). Le même répertoire doit également contenir le fichier binaire exécutable **ENIGMA.EXE** correspondant au programme.