



## Fiche exercice

1NSI

TP

Nom : \_\_\_\_\_

Prénom : \_\_\_\_\_

### 1. Le maximum de trois nombres

Définissez une fonction `maximum(n1,n2,n3)` qui renvoie le plus grand des 3 nombres `n1`, `n2`, `n3` fournis en arguments. Par exemple, l'exécution de l'instruction : `print(maximum(2,5,4))` doit donner le résultat : 5.

### 2. Vérification de l'aléatoire

Créer un algorithme qui demande un nombre `n` à l'utilisateur, puis créer une liste de nombre aléatoire compris entre 0 et 500. Enfin l'algorithme doit vérifier qu'aucun nombre aléatoire soit identiques.

Pour rappel :

Pour créer une liste d'une taille donnée contenant uniquement des 0 vous pouvez utiliser le code suivant :

```
liste = [0]*n
```

### 3. Un tapis magnifique et sur mesure

Écrire une fonction qui, recevant une taille `n` en paramètre, affiche un tapis de `n+1` lignes/`n+1` colonnes traversé par une diagonale.

Pour rappel :

Pour ne pas avoir un saut de ligne après un `print` vous pouvez utiliser le code suivant :

```
print("-",end='')
```

Exemple (pour une taille de 10) :

```
+-----+
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
|#####|
+-----+
```

PS: le "`n+1`" permet de pouvoir gérer plus facilement le cas "0" :

```
+ - +
|   |
+ - +
```



### 4. Le codage César

Jules César, général et stratège romain, a été (à ce qu'il semble) le premier militaire officiel à chiffrer ses messages. Sa méthode était assez simple : il décalait les lettres de 3 rangs dans l'alphabet.

Le but de cet exercice est de créer une fonction à laquelle on donne un message et un décalage, et la fonction renvoie alors le message décalé dans l'alphabet. Il faudra faire attention que le message peut contenir des caractères ne faisant pas forcément partie de l'alphabet et dans ce cas, pour ne pas perdre la signification du texte, ces caractères doivent réapparaître à l'identique dans le message chiffré. De plus, il faudra gérer le dépassement ('z' décalé vers la droite revient sur 'a', et 'a' décalé vers la gauche revient sur 'z').

À noter que la fonction pourra être utilisée aussi bien pour chiffrer que pour déchiffrer (il suffit pour cela de lui passer le message chiffré avec l'opposé du décalage utilisé pour retrouver le message d'origine).