

Exercice :

Soit **T** un tableau rempli par **N** mots (avec $3 \leq N \leq 10$). Les mots sont des chaînes de caractères non vides formées par des lettres alphabétiques. La taille maximale d'un mot est 10 caractères .

On se propose de crypter les mots du tableau **T** en appliquant le principe suivant :

- Chaque lettre d'un mot sera remplacée par la lettre qui la suit par P position dans l'alphabet francais . P est le rang du mot dans le tableau **T**
- On suppose que la lettre qui suit "**Z**" est "**A**" et celui qui suit "**z**" est la lettre "**a**"

Exemple :

Soit le tableau **T** suivant :

Examen	Pratique	En	Informatique
1	2	3	4

Après cryptage **T** devient :

Fybufo	Rtckvkswg	Hp	Mrjsvqexmuyi
--------	-----------	----	--------------

On se propose de crypter et d'afficher le contenu tableau **T** rempli par **N** mots respectant les contraintes décrits ci-dessus

SOLUTION EN PYTHON :

```
from numpy import*
```

```
def saisie ():
    valide=False
    while valide==False:
        n=int(input("donner n"))
        valide=3<=n<=10
    return n
```

```
def remplir (n,t):
    for i in range (n):
        valide=False
        while valide==False:
            t[i]=input("t["+str(i)+"]")
            valide=1<=len(t[i])<=10 and verif(t[i])
```

```
def verif (ch):
    test=True
    for i in range (len(ch)):
        if not("A"<=ch[i]<="Z" or "a"<=ch[i]<="z"):
            test=False
    return test
```

```
def cryptage (n,t):
    for i in range(n):
        ch=""
        for j in range(len(t [ i ])):
            if "A"<=t [ i ][ j ]<="Z":
                r=ord(t [ i ][ j ])-64
                ch=ch+chr(64+(r+i+1)%26)
            else:
                r=ord(t [ i ][ j ])-96
                ch=ch+chr(96+(r+i+1)%26)
        t [ i ]=ch
```

```
def afficher (n,t):
```

```
for i in range(n):  
    print (t[i])
```

```
#p.p
```

```
t=array([str]*10)
```

```
n=saisie()
```

```
remplir(n,t)
```

```
cryptage(n,t)
```

```
afficher(n,t)
```