

## Fiche TP11 Corrigé - Codage texte

Informatique - Culture informatique et syntaxe

### A Codage des caractères

### B Codage UTF-8 et Python

### C Exercices d'application

1. Déterminer le texte associé à la suite de codes 80, 99, 115, 105 et 49.

**Solution :** 'Pcsi1'

2. Ecrire une fonction `en_codes(s)` retournant une liste des codes unicode des caractères de la chaîne de caractères `s`. On peut utiliser une construction par compréhension ou élément par élément. `en_codes('abc')` doit retourner `[97, 98, 99]`.

**Solution :**

```
1 def en_codes(s):
2     return [ord(c) for c in s]
3
4 def en_codes(s):
5     res = []
6     for c in s: # parcours des caractères de s
7         res.append(ord(c))
8     return res
```

Utilisation :

```
> > > en_codes('abc')
```

```
[97, 98, 99]
```

3. Ecrire une fonction `en_texte(L)` retournant une chaîne de caractères. `L` est une liste des codes des caractères à retourner. `en_texte([97, 98, 99])` doit retourner `'abc'`. `en_texte(en_codes(s))` doit retourner la chaîne de caractères `s`. Utiliser une construction élément par élément.

**Solution :**

```
1 def en_texte(L):
2     s = ''
3     for code in L: # parcours des éléments
4         s = s + chr(code)
5     return s
```

Utilisation :

```
> > > L = en_codes('abc')
```

```
> > > en_texte(L)
```

```
'abc'
```

4. On suppose à présent que tous les caractères sont des minuscules, et on souhaite associer le code 0 à la première lettre a, le code 1 à b, le code 2 à c, etc. Modifier les fonction `en_codes` et `en_texte` pour cela. A tester avec des minuscules.

## Solution :

```
1 def en_codes(s):
2     dec = ord('a')
3     return [ord(c)-dec for c in s]
4
5 def en_codes(s):
6     dec = ord('a')
7     res = []
8     for c in s: # parcours des caractères de s
9         res.append(ord(c)-dec)
10    return res
11
12 def en_texte(L):
13     dec = ord('a')
14     s = ''
15     for code in L: # parcours des éléments
16         s = s + chr(code+dec)
17     return s
```

## Utilisation :

```
> > en_codes('abcde')
[0, 1, 2, 3, 4]
> > en_texte([0, 1, 2])
'abc'
```