

## Exercice 1 :

```
Question 1 : print(chr(int("1010101",2))) # Affiche U  
Question 2 : print(bin(int("1010101", 2))) # affiche 0b1010101
```

## Exercice 2 :

```
msg = "1101100 1100101 1110011 100000 1001110 1010011 1001001 100000  
1100011 100111 1100101 1110011 1110100 100000 1101100 1100101 1110011  
100000 1101101 1100101 1101001 1101100 1101100 1100101 1110101 1110010  
1110011"  
liste_codes_ascii = msg.split(" ")  
msg_decode = ""  
for code in liste_codes_ascii:  
    caractere = chr(int(code, 2))  
    msg_decode = msg_decode + caractere  
print(msg_decode)
```

## Exercice 3 :

Sur le site <https://hexed.it/> , taper les valeurs 76 E9 6C 6F puis enregistrer dans un fichier test.txt sur le bureau.  
En ouvrant le fichier, on y voit le mot vélo.

## Exercice 4 :

Unicode apporte un nombre unique pour chaque caractère,  
peut importe la plateforme,  
peut importe le programme,  
peut importe le langage.

## Exercice 5 :

Exemple avec le symbole € :

```
print(ord('€')) # affiche 8364  
print(hex(8364)) # affiche 0x20ac  
# avec 0x20ac, mon caractère sera codé sur 3 octets de la façon suivante :  
# 1110 bbbb 10bb bbbb 10bb bbbb  
print(bin(8364)) # affiche 0b10000010101100  
# On répartit les bits sur les 3 octets  
print(hex(int("11100010", 2))) # affiche 0xe2  
print(hex(int("10000010", 2))) # affiche 0x82  
print(hex(int("10101100", 2))) # affiche 0xac
```

Je vérifie avec hexed.it qu'un fichier contenant E2 82 AC lu en UTF-8 m'affiche bien €.

## Question 1 :

```
print(ord('é')) # affiche 233  
print(bin(233)) # affiche 0b11101001
```

## Question 2 :

```
# On convertit la valeur décimal en hexadécimale pour connaître le nombre d'octets.  
  
print(hex(233)) # affiche 0xe9  
  
# 0xe9 est dans la plage des valeurs de 0x80 à 0x07FF  
# , donc mon caractère sera codé sur 2 octets de la façon suivante 110b bbbb 10bb bbbb.  
# On prend les 6 derniers bits de 0b11101001 pour les mettre dans le 2ème octet : 10bb bbbb → 1010 1001  
# On prend les 2 bits qu'il nous reste de 0b11101001 pour les mettre dans le 1ème octet : 110b bbbb → 110b bb11 et on comble avec des 0. 1100 0011
```

## Question 3 :

```
print(hex(int("11000011", 2))) # affiche 0xc3  
print(hex(int("10101001", 2))) # affiche 0xa9
```

Je vérifie avec hexed.it qu'un fichier contenant C3 A9 lu en UTF-8 m'affiche bien €.

## Question 4 :

On peut utiliser notepad++ pour changer l'encodage, dans Europe de l'Ouest, on choisi ISO 8859-15 et le caractère é est lu comme Â©.

## Question 5 :

L'éditeur de texte ne sait pas quel codage est utilisé dans le fichier, il en essaie un par défaut. Certains caractères sont reconnus car il y a une certaine compatibilité entre les codages (ASCII est inclus dans presque tous les codages) mais les autres caractères sont mal décodés.