

1. Donner la représentation en écriture décimale des nombres suivants :

- a)  $(10101)_2$                       b)  $(10111001)_2$   
c)  $(A5ED)_{16}$                       d)  $(C31)_{16}$

2. Donner la représentation en écriture binaire des nombres suivants :

- a)  $(98)_{10}$                       b)  $(532)_{10}$   
c)  $(CD16)_{16}$                       d)  $(A5ED1)_{16}$

3. Donner la représentation en écriture hexadécimale des nombres suivants :

- a)  $(54310)_{10}$                       b)  $(19810)_{10}$   
c)  $(110000101110)_2$                       d)  $(1001110)_2$

4. Addition binaire

- (a) Écrire la table d'addition en binaire.  
(b) Poser et effectuer l'addition suivante :  $100111001 + 1001111$

5. Pour un zéro de plus...

- (a) Comment est modifié un nombre décimal lorsqu'on ajoute un 0 à sa droite ?  
(b) Comment est modifié un nombre binaire lorsqu'on ajoute un 0 à sa droite ?

6. Couleur RVB

Pour coder une couleur, on utilise souvent le code RVB. Le principe est de donner pour cette couleur l'intensité de ses trois composantes Rouge, Vert, Bleu en hexadécimal ou en décimal. Pour l'intensité, on utilise une échelle allant de 00 à FF en hexadécimal, c'est-à-dire de 0 à 255 en décimal.

Par exemple : la couleur « lilas » est codée (A5 ; 44 ; B9) en hexadécimal ou (165 ; 68 ; 185) en décimal. Ceci signifie qu'en hexadécimal, l'intensité du rouge est A5, celle du vert est 44 et celle du bleu est B9.

- (a) Sur combien d'octets est représentée l'intensité de chaque couleur primaire ?  
(b) Combien de couleurs peut-on représenter avec ce code RVB ?  
(c) La couleur « or » est codée en décimal (255 ; 215 ; 0).  
Déterminer son codage en hexadécimal.  
(d) La couleur « brun » est codée en hexadécimal (5B ; 3C ; 11).  
Déterminer son codage en décimal et en binaire.

7. TP Python :

Vérifier avec Python les exercices précédents.

Les fonctions `bin(n)` et `hex(n)` permettent d'obtenir la représentation d'un entier `n`, respectivement, en binaire et en hexadécimal.

Pour écrire un entier directement en binaire (ou en hexadécimal), on ajoute un préfixe `0b` (ou `0x`).

Écrire une fonction Python `repr_bin` qui donne la représentation d'un entier en binaire. Le nombre binaire sera représenté par une liste-Python de 1 et de 0. Évidemment ici, il est interdit d'utiliser des fonctions intégrées de Python qui « font le travail toutes seules ».

*Exemple d'utilisation :*

```
>>> repr_bin(26)
[1, 1, 0, 1, 0]
>>> repr_bin(0x1A)
[1, 1, 0, 1, 0]
```