NOMS DES ÉLÈVES :

NSI1-1 **Du cours au programme Python** 11-2021

Exercice 1

Rappelle-toi ce que nous avons vu en cours :

Méthode 1 : passer de la base 2 à la base 10

Que vaut $(11101)_2$?

Chiffre binaire	1	1	1	0	1
Valeur	2^4	2^3	2^2	2^1	2^0

$$\begin{aligned}(11101)_2 &= 1 \times 2^4 + 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0 \\ &= 16 + 8 + 4 + 1 \\ &= 29\end{aligned}$$

Tu vas compléter le programme appelé `methode1.py` qui suit… la méthode 1 :

- il demande à l'utilisateur d'entrer un nombre en binaire sous la forme d'une chaine de caractères composées uniquement de 0 et de 1;
- affiche l'écriture décimale du nombre binaire que l'utilisateur a entré.

Comment fonctionne ce programme sur un exemple ?

En reprenant l'exemple de l'encadré on entre `11101` dans une variable `chaine` et

- on voit que la longueur de cette chaine est 5;
- donc `chaine[0]` est le bit de 2^4, `chaine[1]` est le bit de 2^3, …, `chaine[4]` est le bit de 2^0;
- ainsi on peut créer une variable `nombre` qui vaut zéro et une boucle `for` pour parcourir `chaine`;
- si `chaine[i]` vaut 1 on ajoute la valeur correspondante à `somme` sinon on ne fait rien;
- en sortie de boucle on affiche `somme`.

Tu peux déjà commencer par compléter sur papier :

Code Python

```
nombre_binaire = input("Entrez un nombre en binaire :")
valeur = ............
n = len(............)
for i in range(............):
    if nombre_binaire[............] == '1':
        valeur += 2**(............)
print("Cela vaut",valeur)
\end{minted}
```

Ensuite tu peux le programmer avec EDUPYTHON.

Exercice 2

Méthode 2 : passer de la base 10 à la base 2

$$\begin{aligned} 203 &= 128 + 64 + 8 + 2 + 1 \\ &= 2^7 + 2^6 + 2^3 + 2^1 + 2^0 \\ &= 1 \times 2^7 + 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 1 \times 2^3 + 0 \times 2^2 + 1 \times 2^1 + 1 \times 2^0 \\ &= (11001011)_2 \end{aligned}$$

Tu vas compléter le programme `methode2.py` qui

- demande à l'utilisateur un entier positif;
- affiche l'écriture en binaire de cet entier (au format `str`) en suivant la méthode 2;

Comment fonctionne la méthode sur l'exemple ?

- j'ai d'abord déterminé que 128 est la plus grande puissance de 2 inférieure à 203 : je suis parti de 1 puis je l'ai multiplié par 2, par 2 etc, jusqu'à 256. Puisque 256 est strictement plus grand que 203, la plus grande puissance inférieur à 203 est 128.
- j'ai commencé par définir une variable `binaire` de type `str` valant `""`.
- je peux enlever 128 à 203, donc je peux ajouter `"1"` à `binaire` : c'est le bit de 128.
- j'enlève 128 à 203, il reste 75 et je regarde si la puissance de 2 « juste avant » 128 est inférieure à 75 : $64 < 75$ donc à ma variable `binaire` j'ajoute `'1'` (bit de 64).

- je recommence en enlevant 64 à 75 : il reste 11 et 32 « ne rentre pas dans 11 » donc j'ajoute `'0'` à `binaire`.
- et ainsi de suite jusqu'à ce qu'il ne me reste plus rien.

Tu peux commencer par compléter sur papier :

Code Python

```
entier = ............ (input("Entre un entier :"))
print('Je commence par trouver la plus grande puissance de 2
    inférieure à', entier)
i = 0
while 2 ** i <= ............ :

    i = i + 1
i = i ............


binaire = ............
while entier > 0:

    if entier >= ............ :
        binaire = binaire + "1"
        entier = entier - 2 ** i
    else:
        binaire += ............
    i = ............

print("En définitive j'ai trouvé", binaire)
```

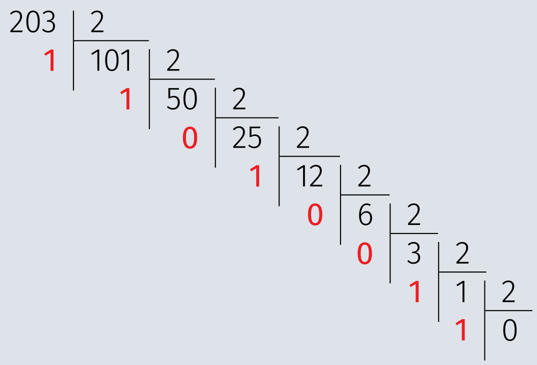
Programme ensuite ton script avec `EduPython`.

Exercice 3

Tu vas devoir compléter le programme `methode3.py` qui utilise la méthode des divisions successives par 2 pour obtenir l'écriture binaire (au format `str`) d'un entier.

Méthode 3 : les divisions successives

Voici comment on trouve les chiffres de l'écriture *binaire* de 203 :



En définitive, **203 = $(11001011)_2$**.

Pour cet exercice il faut se « débrouiller tout·e seul·e » en tirant les leçons des exercices précédents.

1. Écrire un programme à la main qui :
   - demande un entier positif à l'utilisateur ;
   - affiche son écriture en binaire en appliquant la méthode précédente.
2. Écrire le programme Python sur l'ordinateur, il devra s'appeler `methode3.py`

Tu peux commencer par compléter sur papier :

Code Python

```
entier = ............ (input("Entrez un nombre :"))

resultat = ............
while entier != ............ :
    resultat = str(............) + resultat
    entier = entier // 2
print(resultat)
```

Programme ensuite ton script avec `EduPython`.