# Convertisseur binaire / décimal

# **6** Objectif

L'objectif de ce projet est de programmer un convertisseur d'écritures:

- d'un nombre décimal en binaire;
- d'un nombre binaire en décimal.

On n'utilisera bien entendu pas les fonctions natives de Python pour effectuer ces conversions.

Il faudra écrire des **fonctions** qui effectuent ces conversions.

Décimal - Binaire

- la fonction prendra en paramètre un nombre entier (type int);
- la fonction renverra sa conversion en binaire sous forme d'une chaîne de caractères (type str ).

## Pour cela:

- · utiliser l'algorithme des divisions successives;
- stocker les restes dans une chaîne de caractères (attention à l'ordre)

Binaire - Décimal

- la fonction prendra en entrée un nombre binaire sous forme d'une chaîne de caractères (type str);
- la fonction renverra sa conversion en décimal sous forme d'un nombre entier (type int ).

# Pour cela:

- parcourir les caractères du nombre binaire;
- incrémenter de la bonne puissance de deux un int selon les valeurs des caractères (convertis en int).

#### Ouverture

Réaliser de même un convertisseur hexadécimal / décimal.

# Proposition de correction

**Décimal** → **Binaire** 

## **%** Script Python

```
1
     def decimal_vers_binaire(n: int) -> str:
         1.1.1
 2
         Renvoie la conversion de l'entier n en mot binaire,
 3
         en utilisant l'algorithme des divisions successives.
 4
         - précondition: n doit être un entier positif.
 5
         1 1 1
 6
 7
         assert n \ge 0
         mot = ''
 8
 9
         while n != 0:
10
             mot = str(n\%2) + mot
11
             n = n // 2
12
         return mot
```

Binaire - Décimal

#### **%** Script Python

```
1
     def binaire_vers_decimal(b: str) -> int:
         1.1.1
 2
         Renvoie l'écriture décimale d'un nombre donné par son mot binaire,
 3
         en utilisant la définition de la base 2.
 4
 5
 6
         exposant = len(b) - 1
         decimal = 0
 7
8
         for bit in b:
             decimal = decimal + int(bit)*2**exposant
 9
10
             exposant -= 1
11
         return decimal
```