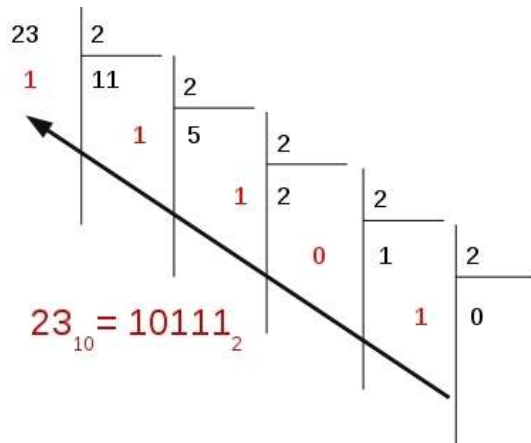


## 1. Conversion décimale vers binaire

Pour convertir un nombre entier positif N de la base décimale à la base binaire, il faut opérer par des divisions successives du nombre N par 2. Les restes des divisions constituent la représentation binaire.

Ex :



On arrête les divisions par deux quand le quotient devient nul.

On lit alors les reste de division dans l'ordre inverse de leur calcul.

Pour arriver au dernier reste on a effectué n division d'où le terme puissance croissante.

### 1.1. Conversion décimale vers binaire itératif

Coder la fonction « conversion\_dec\_bin »

#### Speudo code

```

fonction conversion_dec_bin(n)
    calculer le quotient de n par 2
    calculer le reste de n par 2
    initialiser une chaine de caractère vide
    Si quotient == 0 alors
        Remplir la chaine avec le reste
    Retourner la chaine
    Sinon
        Tant que le quotient est différent de 0
            calculer le quotient de n par 2
            calculer le reste de n par 2
            affecter à n le quotient précédemment calculé
            ajouter à la chaine de reste
        Inverser la chaine de caractères
    Retourner la chaine
  
```

## 1.2. Conversion décimale vers binaire récursif

Le tant que de la version itérative calcul un quotient puis remplace n par ce même quotient.

On va utiliser cette propriété en créant une fonction `conversion_dec_bin_recuratif(n)`

```

fonction conversion_dec_bin_recuratif(n)
    cas n == 0
        retourner « 0 » ( Le caractère 0 )
    autre
        s'occuper du cas précédent + ne pas oublier le reste
        retourner le tout sous forme de chaîne de caractère

```

## 2. Conversion décimale vers hexadécimal

S'inspirer du binaire pour la conversion décimal vers hexadécimal. On utilisera avec profit la fonction suivant de conversion pour un seul nombre hexadécimal :

```

def dec_vers_hexa(n):
    if n == 0 :
        return '0'
    elif n == 1 :
        return '1'
    elif n == 2 :
        return '2'
    elif n == 3 :
        return '3'
    elif n == 4 :
        return '4'
    elif n == 5 :
        return '5'
    elif n == 6 :
        return '6'
    elif n == 7 :
        return '7'
    elif n == 8 :
        return '8'
    elif n == 9 :
        return 9
    elif n == 10:
        return 'A'
    elif n == 11 :
        return 'B'
    elif n == 12 :
        return 'C'
    elif n == 13:
        return 'D'
    elif n == 14 :
        return 'E'
    elif n == 15 :
        return 'F'

```

