Exercice : écriture binaire

```
In [1]: def binaire(n:int)->str:
        """Conversion d'un nombre décimal en une chaîne de caractère représentant n en binaire"""
        assert isinstance(n,int), "l'entrée n'est pas décimale"
        binstr=''
        if n==0:
            return '0'
        while n>0:
            binstr=str(n%2)+binstr
            n=n//2
        return binstr
```

```
In [2]: binaire(8), binaire(13), binaire(0)
```

```
Out[2]: ('1000', '1101', '0')
```

```
In [3]: def decimal(c:str)->int:
        """Conversion d'une chaîne de caractères représentant un nombre en binaire en un entier"""
        assert isinstance(c,str), "l'entrée n'est pas une chaîne de caractère"
        n=0#initialisationde l'entier à retourner
        N=len(c)# nombre de bit dans la chaîne de caractères
        for k in range(len(c)):
            n=n+int(c[k])*2**(N-1-k)
        return (n)
```

```
In [4]: decimal('1000'),decimal('1101'),decimal('0')
```

```
Out[4]: (8, 13, 0)
```

```
In [5]: def binairesigne(n:int,N:int)->str:
        """Conversion d'un entier en représentation binaire signée sur Nbits"""
        assert isinstance(n,int) and isinstance(N,int), "les entrées ne sont pas des entiers"
        if 0<=n<2**(N-1):
            res=binaire(n)
            res=(N-len(res))*'0'+res
        elif n>2**(N-1)-1 or n<-2**(N-1):
            res='pas codable sur N bits'
        else:# nombre négatif entre -2**(N-1 et -1 inclus)
            n=n+2**N
            res=binaire(n)
        return res
```

```
In [6]: binairesigne(8,5),binairesigne(-7,5),binairesigne(0,5),binairesigne(35,5)
```

```
Out[6]: ('01000', '11001', '00000', 'pas codable sur N bits')
```

```
In [7]: def decimalsigne(c:str,N:int)->int:
        """Donne l'entier relatif codé sur N bits par la chaîne de caractères c"""
        assert isinstance(c,str) and isinstance(N,int), "les entrées ne sont pas aux bons format"
        n=decimal(c)
        if n<2**(N-1):# Cas positif
            return n
        elif 2**(N-1)<=n<2**N:# Cas négatif
            return n-2**N
        else:
            return None
```

```
In [8]: decimalsigne('01000',5), decimalsigne('11001',5), decimalsigne('00000',5), decimalsigne('10000',4)
```

```
Out[8]: (8, -7, 0, None)
```

Exercice : nombre de uns

```
In [9]: def binaire(n:int)->list:
        """Convertit un entier naturel en une liste contenant son codage en binaire"""
        L=[]
        while n!=0:
            L=[n%2]+L
            n=n//2
        return L
```

```
In [10]: binaire(23)
```

```
Out[10]: [1, 0, 1, 1, 1]
```

```
In [11]: def NombreDeUns(n:int)->int:
        """Compte le nombre de un dans la représentation binaire d'un entier naturel"""
        L=binaire(n)
        s=0
        for k in range(len(L)):
            s=s+L[k]
        return s
```

```
In [12]: NombreDeUns(23)
```

```
Out[12]: 4
```