

# TP : conversion binaire et rendu de monnaie

## Exercice 1

On modélise la représentation binaire d'un entier non signé par un tableau d'entiers dont les éléments sont 0 ou 1. Par exemple, le tableau [1, 0, 1, 0, 0, 1, 1] représente l'écriture binaire de l'entier dont l'écriture décimale est

$2^6 + 2^4 + 2^1 + 2^0 = 83$ . À l'aide d'un parcours séquentiel, écrire la fonction convertir répondant aux spécifications suivantes :

```
1 def convertir(T):
2     '''
3     T est un tableau d'entiers, dont les éléments sont 0 ou 1 et
4     représentant un entier écrit en binaire. Renvoie l'écriture
5     décimale de l'entier positif dont la représentation binaire
6     est donnée par le tableau T
7     '''
```

Exemple :

```
1 >>> convertir([1, 0, 1, 0, 0, 1, 1])
2 83
3 >>> convertir([1, 0, 0, 0, 0, 0, 1, 0])
4 130
```

## Exercice 2

La fonction rendu\_monnaie\_centimes prend en paramètres deux nombres entiers positifs s\_due et s\_versee et elle permet de procéder au rendu de monnaie de la différence s\_versee - s\_due pour des achats effectués avec le système de pièces de la zone Euro. On utilise pour cela un algorithme qui commence par rendre le maximum de pièces de plus grandes valeurs et ainsi de suite. La fonction renvoie la liste des pièces qui composent le rendu.

Toutes les sommes sont exprimées en centimes d'euros. Les valeurs possibles pour les pièces sont donc [1, 2, 5, 10, 20, 50, 100, 200].

Ainsi, l'instruction rendu\_monnaie\_centimes(452, 500) renverra la liste [20, 20, 5, 2, 1]. En effet, la somme à rendre est de 48 centimes soit  $20 + 20 + 5 + 2 + 1$ .

Le code de la fonction est donné ci-dessous :

```
1 def rendu_monnaie_centimes(s_due, s_versee):
2     pieces = [1, 2, 5, 10, 20, 50, 100, 200]
3     rendu = ...
4     a_rendre = ...
5     i = len(pieces) - 1
6     while a_rendre > ... :
7         if pieces[i] <= a_rendre :
8             rendu.append(...)
9             a_rendre = ...
10        else :
11            i = ...
12    return rendu
```

Compléter ce code pour qu'il donne :

```
1 >>> rendu_monnaie_centimes(700,700)
2 []
3 >>> rendu_monnaie_centimes(112,500)
4 [200, 100, 50, 20, 10, 5, 2, 1]
```