

```
"""
Author: Pascal Padilla
Source: correction de l'exercice 1 du sujet 13 des épreuves pratiques NSI 2022
"""
```

```
from doctest import testmod
```

```
def rendu(somme_a_rendre):
    """Rendu de monnaie sous forme de tableau contenant
    le nombre de billets de 5€, de pièces de 2€ et de 1€

    Args:
        somme_a_rendre (int): valeur dont on doit faire la monnaie

    Returns:
        list: tableau de 3 cases contenant :
            * le nb de billets de 5€
            * le nb de pièces de 2€
            * le nb de pièces de 1€

    Tests et exemples:
    >>> rendu(13)
    [2, 1, 1]
    >>> rendu(64)
    [12, 2, 0]
    >>> rendu(89)
    [17, 2, 0]
    """
    #####
    # Première méthode #
    # avec trois boucles #
    #####

    # somme d'argent qu'il reste à rendre
    reste = somme_a_rendre

    # BOUCLE
    # invariant de boucle:
    # nb_5_euros × 5 + reste = somme_a_rendre
    # condition d'arrêt:
    # reste < 5
    nb_5_euros = 0
    while not reste < 5:
        nb_5_euros += 1
        reste -= 5

    # BOUCLE
    # invariant de boucle:
    # nb_2_euros × 2 + nb_5_euros × 5 + reste = somme_a_rendre
    # condition d'arrêt:
    # reste < 2
    nb_2_euros = 0
    while not reste < 2:
        nb_2_euros += 1
        reste -= 2

    # la somme qui reste est égale au nb de pièces de 1€ ;)
    nb_1_euros = reste

    # return [nb_5_euros, nb_2_euros, nb_1_euros]

    #####
    # Deuxième méthode #
    # avec 3 divisions euclidiennes #
    #####
    reste = somme_a_rendre

    # le nombre de billets de 5 € est égal au quotient de la
    # division euclidienne de ce qui reste à rendre par 5
    nb_5_euros = reste // 5

    # le reste de la monnaie à rendre est égal
```

```
#    au reste de la division euclidienne
reste = reste % 5

# idem avec le nb de pièce de 2 euros
nb_2_euros = reste // 2
reste = reste % 2

# la somme qui reste est égale au nb de pièces de 1€ ;)
nb_1_euros = reste

return [nb_5_euros, nb_2_euros, nb_1_euros]

# tests de l'énoncé
testmod()
```