

Ollivander :

VARIABLES

```
monnaie ← liste  
monnaie_rendue ← liste  
monnaie_rendue ← [0, 0, 0]
```

Fonction ollivander (monnaie):

```
monnaie_rendue[1] ← monnaie[2] // 29 + monnaie[1]  
monnaie_rendue[2] ← reste de monnaie[2] par 29  
monnaie_rendue[0] ← monnaie_rendue[1] // 17 + monnaie[0]  
monnaie_rendue[1] ← reste de monnaie_rendue[1] par 17
```

Renvoyer monnaie_rendue

Fleury et Bott :

VARIABLES

```
monnaie ← entier  
monnaie_rendue ← dictionnaire  
monnaie_dispo ← tuple
```

Fonction flourish_and_blotts(monnaie) ← dictionnaire:

vérifier que monnaie est un entier, afficher "Vos devez rentrer un nombre conforme"

```
monnaie_dispo ← (500, 200, 100, 50, 20, 10, 5, 2, 1)  
monnaie_rendue ← {500 : 0, 200 : 0, 100 : 0, 50 : 0, 20 : 0, 10 : 0, 5 : 0, 2 : 0, 1 : 0}
```

POUR billet dans monnaie_dispo:

```
monnaie_rendue[billet] ← monnaie // billet  
monnaie ← monnaie % billet
```

FIN POUR

Renvoyer monnaie_rendue

Guipure :

VARIABLES

```
monnaie_dispo ← {200: 1, 100 : 3, 50: 1, 20: 1, 10: 1, 2: 5}  
rendu_caisse ← {200: 0, 100: 0, 50 : 0, 20: 0, 10: 0, 2: 0}
```

POUR billet DANS monnaie_dispo :

```
TANT QUE rendu ≤ billet ET monnaie_dispo[billet] > 0:  
    monnaie_dispo[billet] ← monnaie_dispo[billet] -1  
    rendu ← rendu - billets  
    rendu_caisse[billet] ← rendu_caisse[billet] + 1
```

FIN TANT

FIN POUR

Renvoi de la valeur rendu_caisse