Le glouton

1. Le principe

On veut réaliser une fonction ***renduMonnaie(somme,pieces)*** qui détermine les pièces à rendre dans un monnayeur.

|  |  |
| --- | --- |
| Rendre la somme de 8€ | Solution |
|  |  |

1. Codage de l'algorithme ♥

|  |  |
| --- | --- |
| **Algorithme pseudo code** | **Python** |
| Fonction renduMonnaie (somme en entier, pièces : liste des pièces du monnayeur dans l’ordre décroissant) : dictionnaire des pièces choisies  initialiser à zéro le dictionnaire choisies Pour p dans pieces  choisies[p]← 0  Tant que somme>= p  somme←somme-p  choisies[p]← choisies[p]+1  fin tant que  fin pour  retourner choisies |  |

|  |  |
| --- | --- |
| **Résultat dans la console** | |
| *#pieces en centimes d'euros*  pieces=[500,200,100,50,20,10,5,2,1]  somme=780  print(**'Les pièces choisies sont'**) print(renduMonnaie(somme,pieces)) | Les pièces choisies sont |

En utilisant une seule boucle pour :

|  |  |
| --- | --- |
| **Algorithme pseudo code** | **Python** |
| Fonction renduMonnaie (somme en entier, pièces : liste des pièces du monnayeur dans l’ordre décroissant) : dictionnaire des pièces choisies  initialiser à zéro le dictionnaire choisie Pour p dans pieces  nb← somme division entière par p choisies[p]← nb  somme←somme – nb \*p  fin pour  retourner choisies |  |

Remarque : C'est un algorithme très simple et rapide, et on appelle canonique **un système de pièces pour lequel cet algorithme donne une solution optimale quelle que soit la valeur à rendre**.