

## Interrogation - Informatique

Pour partir un week-end à la montagne, on dispose d'un sac à dos pouvant contenir jusqu'à 4 kg. Il s'agit de le remplir au mieux. Pour cela, on établit une liste d'objets utiles. On attribue à chaque objet une note qui mesure son utilité. Le tableau ci-dessous présente cette liste d'articles et indique le poids (en grammes), la note et la quantité disponible de chaque article.

item	weight (g)	value	piece(s)
map	90	150	1
compass	130	35	1
water	1530	200	2
sandwich	500	60	2
glucose	150	60	2
tin	680	45	3
apple	390	40	3
cheese	230	30	1
suntan cream	110	70	1
camera	320	30	1
T-shirt	240	15	2
trousers	480	10	2
umbrella	730	40	1
waterproof trousers	420	70	1
waterproof overclothes	430	75	1
note-case	220	80	1
sunglasses	70	20	1
towel	180	12	2
socks	40	50	1
book	300	10	2

On suppose que le tableau est stocké dans un dictionnaire `DicArticles`. Les clés sont les noms des articles, les valeurs sont des listes contenant le poids, la valeur et le nombre correspondant à la clé.

Par exemple la valeur de la clé `sandwich` est la liste `[500,60,2]`.

1. On veut transformer le dictionnaire en liste `ListeArticles` dans laquelle chaque élément est une liste contenant le nom de l'article, son poids, sa valeur et sa quantité.

Par exemple si le premier élément de la liste est `map`, la valeur de `ListeArticles [0]` doit être `['map',90,150,1]`.

Écrire un script qui crée la liste `ListeArticles`.

2. Avant d'appliquer l'algorithme glouton, on trie les articles par ordre décroissant de leur valeur par unité de poids :

```
ListeArticles .sort(key=valeurrelative , reverse = True)
```

Dans cette instruction la fonction `valeurrelative` renvoie le quotient de la valeur sur le poids de l'article. Cette fonction prend en argument une liste `A` représentant un article (par exemple `['map',90,150,1]`).

Écrire la fonction `valeurrelative`.

3. Recopier et compléter l'algorithme glouton ci-dessous dans lequel la liste `L` contient (avec répétition) les numéros des articles que l'on place dans le sac à dos.

```
def glouton(poidsmax):
    poids=0
    N=0
    L=[]
    while N<len(ListeArticles):
        if _____
            L.append(N)
            ListeArticles [N][3]= _____
            poids= _____
        else:
            _____
    return L,poids
```

4. Écrire une fonction qui donne, sous forme d'un dictionnaire, la composition du sac à dos à partir de la liste donnée par l'algorithme glouton.