

TP sac à dos

On a un sac à dos à remplir, qui a une limite de masse **mMax** (on prendra ici 15 kg).

Les objets qui peuvent y prendre place ont leur masse **m_i** et une valeur **x_i** (qui donne une mesure de l'importance de ces objets).

On veut obtenir le *meilleur* sac à dos possible.

Données : sacADos.py

Tout les objets à prendre sont dans la variable *mesObjets* :

On utilise un dictionnaire donc la clé est le nom de l'objet et la valeur est une liste composée de la masse de l'objet et de la valeur (ne mélangez pas valeur-value du dictionnaire et les valeurs associées l'importance de chaque objet).

Q1 : complète la fonction *afficheObjets(d)* pour avoir l'affichage de chaque objet avec masse et valeur.

Exemple : *tente de masse 7 kg et de valeur 50*

Principe

On désire remplir le sac à dos afin qu'il y ait une valeur la plus grande possible sans dépasser 15 kg.

Cependant, il y a plusieurs manières de considérer l'ordre dans lequel placer les objets : soit en considérant uniquement la masse, soit en considérant uniquement la valeur, soit en essayant de prendre en compte les deux grandeurs.

Une fois l'ordre défini et le tri effectué on aura plus qu'à remplir le sac à dos sans dépasser 15 kg (on voit se profiler une boucle while).

L'algorithme est donc relativement simple, il faut maintenant trier (déjà-vu mais là on a des dictionnaires).

On va expérimenter les 3 critères afin d'obtenir dans les trois cas **les objets à placer** et la **valeur totale**.

Premier critère : suivant le poids

Sur le critère du poids-masse, il s'agit pour avoir la plus grande valeur possible de placer les objets dans l'ordre croissant de leur masse afin d'en avoir le plus possible en nombre et ainsi une valeur importante.

Le problème est la notion d'ordre sur un dictionnaire et de manipuler ensuite les données.

On peut s'arranger pour reconstruire une liste de listes regroupant toutes les données et ensuite trier avec la méthode `sort()` en modifiant les listes pour que le tri se fasse suivant le critère voulu.

Ici, on va utiliser la méthode `sorted()` qui peut prendre en paramètre une fonction *lambda* imposant une manière de trier.

Q2 : récupère le code ci-dessous et place le dans le fichier *sacADos.py* et teste-la. Quelle est la nature du résultat ?

```
def tri1(d):  
    trier=sorted(d.items(), key=lambda t: t[1][0])  
    return trier
```

Q3 : explique ce qui se passe si on met - devant $t[1][0]$

Q4 : écrire une fonction *algo1(l)* permettant d'afficher les objets à placer dans le sac à dos suivant ce premier critère et donnant la valeur totale (on doit trouver une valeur totale de 151).

Deuxième critère : suivant la valeur

Si on choisit les objets suivant leur valeur, il faut trier celle-ci par ordre décroissant.

Q5 : En utilisant un code similaire à *tri1*, complète la fonction *tri2*.

Q6 : écrire la fonction *algo2(l)* permettant d'afficher les objets à placer dans le sac à dos suivant ce deuxième critère et donnant la valeur totale (on doit trouver une masse totale de 14,9 kg). Remarque.

Troisième critère : le rapport **valeur/masse**

Q7 : Insère la fonction suivante dans *sacADos.py* et applique-la à *mesObjets*. Explique le résultat

```
def dic_rapport(d):  
    for e in d.keys():  
        d[e]=d[e]+[d[e][1]/d[e][0]]  
    return d
```

Q8 : Avec des fonctions *tri3(d)* et *algo3(l)*, donne la valeur obtenue en utilisant comme critère de choix le rapport **valeur/masse**

Conclure...