

# Exercice- Algorithmes Gloutons

Pour partir en randonnée, vous disposez d'un sac à dos. Afin de pouvoir marcher longtemps chaque jour, vous décidez de limiter la masse totale transportée à 17kg. Cette limitation ne vous permet pas de pouvoir transporter tous les éléments que vous aimerez emporter.

Vous décidez de donner une note "utilité" à chacun des objets.

Objet	eau	nourriture	nécessaire de cuisine	couverture	matelas	tente	gaz	lampe	console de jeu	change	jumelle	cartes	pull	machette	livre de chevet
Masse	3	4.3	2.2	1	1.5	3.4	0.8	0.5	2.1	2.8	1.3	0.5	0.8	2.4	0.3
Utilité	20	15	12	6	4	18	13	10	7	11	9	8	8	5	2

Vous cherchez à maximiser l'utilité totale des objets emportés dans votre sac à dos.

## Sans ordinateur

1. Le fait de prendre ou non chaque objet est un choix binaire : on peut prendre 3 litres d'eau ou pas, une couverture ou pas, etc. Si vous voulez tester toutes les possibilités pour remplir votre sac à dos afin de trouver la ou les solutions optimales, combien de cas différents devriez-vous étudier ?
2. Si vous utilisez un algorithme glouton avec comme critère l'optimisation de la masse, c'est-à-dire en faisant en sorte de mettre d'abord l'objet le plus lourd, puis le deuxième le plus lourd, etc. sans jamais remettre en cause vos choix précédents, c'est-à-dire sans jamais enlever du sac un objet déjà mis, quelle sera la composition de votre sac à dos si vous limitez la masse totale à 17 kilogrammes ?
3. Si vous utilisez un algorithme glouton avec désormais comme critère l'optimisation de l'utilité, quelle sera la composition de votre sac à dos ? (La masse totale reste encore limitée à 17 kg)

## Avec ordinateur

N'étant pas certain.e.s de la pertinence du résultat obtenu, vous décidez de choisir les objets non plus en fonction de leur seule utilité mais en fonction de leur **utilité massique** c'est-à-dire en fonction du rapport  $\frac{\text{utilité}}{\text{masse}}$ .

Par exemple, l'utilité massique de l'eau est de  $\frac{20}{3} \approx 6.666666666666667$ .

De plus, plutôt que de tout recalculer à la main, vous préférez utiliser une programme informatique en langage Python.

L'ensemble des objets pouvant être choisis pour la randonnée est implémentée par une liste de dictionnaires, chaque dictionnaire représente un objet et a trois clés : `"Nom"` associé au nom de l'objet, `"Masse"` associée à la masse de l'objet et `Utilite` associée à son utilité.

Par exemple, le dictionnaire associé à l'objet eau doit devenir `{'Nom': 'eau', 'Masse': 3, 'Utilite': 20}`.

4. Proposer un script Python qui permet de compléter chaque dictionnaire de la liste en rajoutant une association dont la clé est la chaîne de caractères `"Utilite_massique"` et dont la valeur associée est l'utilité massique de l'objet considéré.  
Par exemple, le dictionnaire associé à l'objet eau doit devenir `{'Nom': 'eau', 'Masse': 3, 'Utilite': 20, 'Utilite_massique': 6.666666666666667}`.
5. Quel prétraitement sur la liste de dictionnaires est-il pertinent d'effectuer afin de pouvoir écrire un algorithme glouton renvoyant un résultat selon le critère pris en compte ?
6. Pour trier une liste de dictionnaires `liste_a_trier`, du type de ceux de cet exercice, par ordre croissante de la masse, il suffit de saisir :  
`liste_triee = sorted(liste_a_trier, key=lambda dico : dico["Masse"])`.  
Proposer une implémentation en langue Python de l'algorithme glouton qui à partir de la liste des objets disponibles affiche la liste des objets à prendre en maximisant le critère de l'utilité massique.