? À rendre mercredi 3 octobre 2025

Devoir Maison n°2 : Dictionnaires et compagnies

- L'objectif de ce DM est de vous faire reprendre en main le langage python.
  Faites ce DM sérieusement, seul.e, en vous aidant de vos cours/TP de PCSI/MPSI.
  Cela vous permettra de faire le point sur ce qu'il vous reste ou non de 1re année, et pour moi d'identifier vos points forts, difficultés...
- Le DM devra être rendu en **manuscrit**. Je vous encourage à y réfléchir dessus avec un crayon et un papier, sans ordinateur.
- Pour toute question, n'hésitez pas à me contacter par mail : nvalade.pcsi@gmail.com

Travail à faire :

♪ Si vous n'êtes pas/peu à l'aise en informatique : Exercice n°1

♪ ♪ Si vous êtes à l'aise en informatique : Exercice n°2

I Rappels de 1re année

Reprendre vos cours/TD de MPSI/PCSI sur les dictionnaires.
Travailler le polycopié de cours « Chapitre n°2. Les dictionnaires » partie I

II Exercices

Exercice n°1 Fonctions élémentaires ♪

Q1. On suppose disposer d'un dictionnaire qui contient les températures à Montélimar à 8h00.
Un exemple et extrait de ce dictionnaire est donné ici :

```
{'J1':-10,'J2':-9,'J3':-4,'J4':0,'J5':-1,'J6':4,'J7':-5,'J8':1,'J9':-2}
```

Écrire une fonction `moyenne(d:dict)->float` qui prend en argument un dictionnaire `d` du type de celui défini ci-dessus et qui renvoie la valeur moyenne des températures.

Q2. Étant donné un dictionnaire dont les clés sont les noms des élèves et les valeurs sont les listes des notes obtenus par l'élève. Écrire la fonction qui renvoie un autre dictionnaire dont les clés sont le nom des élèves et les valeurs la moyenne de leur note `nom_moy(d:dict)->dict`.
Par exemple :
`nom_moy({'Louise':[12,15,12],'Gabriel':[15,17,16],'Léon':[8,18,10],'Alex':[9,14,13]})` renverra `{'Louise':13,'Gabriel':16,'Léon':12,'Alex':12}`.
Indication : On pourra commencer par programmer une fonction intermédiaire.

Q3. Écrire la fonction `mind(d:dict)->float` qui prend en argument un dictionnaire du type du précédent, et qui renvoie la valeur minimale.
On pourra utiliser `float('inf')` qui donne « l'infini ».

Q4. Écrire la fonction `cle_min(d:dict)->str` qui prend en argument un dictionnaire du type du précédent, et qui renvoie la clé de valeur minimale.

On appelle `nb_oc(L:list[int])->dict` une fonction prenant comme argument une liste d'entiers naturels `L` et renvoyant un dictionnaire `D` dont les clefs sont les éléments de `L` et la valeur associée est le nombre d'occurrences de cet élément dans `L`.

Q5. Que renvoie la commande `nb_oc([4,0,0,1,2,0,4])` ?

Q6. Écrire en Python cette fonction `nb_oc(L:list[int])->dict`.

Exercice n°2 Suite de Robinson ♪ ♪

On appelle `nb_oc(L:list[int])->dict` une fonction prenant comme argument une liste d'entiers naturels `L` et renvoyant un dictionnaire `D` dont les clefs sont les éléments de `L` et la valeur associée est le nombre d'occurrences de cet élément dans `L`.

Q1. Que renvoie la commande `nb_oc([4,0,0,1,2,0,4])` ?

Q2. Écrire en Python cette fonction `nb_oc(L:list[int])->dict`.

Q3. Déterminer la complexité de votre fonction `nb_oc(L:list[int])->dict`. Quelle est sa complexité ? Justifier.

Q4. Si votre fonction `nb_oc` n'était pas de complexité linéaire, la réécrire afin qu'elle le soit.

Soit A une liste d'entiers naturels. On définit la suite de Robinson $(L_n)_{n\in\mathbb{N}}$ associée à la suite A par récurrence comme suit :

- $L_0 = A$.
- Si L_n est construite, alors :
  - on détermine $T_n = \texttt{nb_oc}\,(L_n)$.
  - on détermine I_n la liste des clefs non nulles de T_n classée par ordre strictement croissant.
  - si $I_n = [i_1, \ldots, i_r]$, alors $L_{n+1} = [T\,[i_r]\,, i_r, \ldots, T\,[i_1]\,, i_1]$.

Par exemple si $A = [4, 4, 1, 2]$, alors :

- $L_0 = [4, 4, 1, 2]$
- $L_1 = [2, 4, 1, 2, 1, 1]$ (il y a deux « 4 », un « 2 » et un « 1 » dans la liste L_0).
- $L_2 = [1, 4, 2, 2, 3, 1]$ (il y a un « 4 », deux « 2 » et trois « 1 » dans la liste L_1).

Q5. On donne $A = [2, 0, 4, 1, 3, 3, 2, 3, 1, 1]$. Déterminer L_3 et L_{2024}.

Q6. On donne $B = [2, 4, 1, 1, 1, 2]$, on suppose que $L_1 = B$, donner toutes les solutions possibles pour L_0.

Q7. On donne $C = [2, 4, 1, 0]$. Si l'on suppose que $L_1 = C$, donner toutes les solutions possibles pour L_0.

Q8. Proposer alors une fonction `rob(A:list[int], n:int)->list[int]` qui prend en arguments une liste d'entiers naturels `A` et un entier naturel `n` et qui renvoie l'élément L_n de la suite de Robinson associée à `A`.