



Lecture de données dans un fichier

I. Révisions sur les listes

Question 1 : Écrire une fonction **moyenne (L)** qui calcule et retourne la moyenne des termes de la liste L . Donner un ordre de grandeur du temps d'exécution en fonction de n , où n est la taille de la liste L .

Question 2 : On considère la fonction :

```
def maximum(L) :  
    m=L[0]  
    i=1  
    while i<len(L) :  
        if L[i]>m:  
            m=L[i]  
        i=i+1  
    return m
```

- (a) Expliquer ce que représente le résultat renvoyé par **maximum(L)** où **L** est une liste de nombres.
- (b) Donner un ordre de grandeur du temps d'exécution en fonction de n (taille de la liste **L**).
- (c) Proposer un invariant de boucle permettant de démontrer que le fonctionnement de cette fonction est correct.

Question 3 : Écrire une fonction **imaximum(L)** qui renvoie un indice $m \in \llbracket 0, n - 1 \rrbracket$ tel que $L[m]$ soit le maximum de $L[0], \dots, L[n - 1]$.

II. Calculs sur des données contenues dans un fichier

On a enregistré dans un fichier nommé **rfxcmd.txt** des mesures de température effectuées à différents instant par plusieurs sondes. On donne ci dessous un extrait de ce fichier :

```
2,1392837237,FE02,20.7
4,1392837278,FE02,20.7
6,1392837319,FE02,20.7
9,1392837360,FE02,20.7
10,1392837364,B904,20.5
12,1392837401,FE02,20.7
```

```
13,1392837407,B904,21
15,1392837442,FE02,20.7
16,1392837450,B904,21.2
17,1392837465,9C01,20.6
...
```

Chaque ligne de ce fichier est constituée d'un identifiant (nombre entier), d'une date (également sous la forme d'un nombre entier), de la référence de la sonde ayant effectué la mesure et enfin de la mesure de température proprement dite. Ces informations sont séparées par des virgules.

Pour répondre aux questions suivantes, on pourra se reporter à l'annexe 1 (rappels sur les manipulations de fichiers) et l'annexe 2 (présentation de la commande **split**).

Question 4 : Écrire les commandes permettant d'ouvrir le fichier, parcourir toutes les lignes et enregistrer dans une liste **T** toutes les mesures de température ainsi obtenues.

Question 5 : Écrire les commandes permettant d'afficher la moyenne des températures contenues dans le fichier.

Question 6 : Écrire les commandes permettant d'afficher les moyenne des températures contenues dans le fichier et concernant uniquement la sonde **B904**.

Question 7 : Écrire les commandes permettant d'enregistrer dans un fichier **rfxmoy.txt** les moyennes des sondes **B904** et **FE02** sous la forme :

B904,
FE02,

III. Utiliser une base de données

Question 8 : Donner le schéma d'une table qui pourrait contenir les informations enregistrées dans le fichier **rfxcmd.txt**

Question 9 : Donner des commandes SQL permettant d'obtenir les réponses aux question 5 et 6 en utilisant la table de la question précédente.

--

Annexe 1 : écriture et lecture dans un fichier

Quelques exemples en utilisant la console PYTHON :

```
>>> fichier=open('rmv-exemple.txt','w')
>>> fichier.writelines(["Un exemple\n","de\n","fichier\n"])
>>> fichier.close()
```

Remarques.

- **rmv-exemple.tex** est le nom du fichier, **'w'** signifie que l'on ouvre un nouveau fichier pour écrire des informations dedans (si le fichier existe déjà, le contenu sera effacé).
- Les chaînes de caractères que l'on écrit dans le fichier se terminent par **'\n'** qui est le caractère de retour à la ligne.
- Le contenu du fichier est alors :

```
Un exemple
de
fichier
```

```
>>> fichier=open('rmv-exemple.txt','r')
>>> L=fichier.readlines()
>>> fichier.close()
>>> print L
['Un exemple\n', 'de\n', 'fichier\n']
```

Annexe 2 : la fonction `split`

Cette fonction permet de découper une chaînes de caractères en utilisant un *séparateur*.
Exemple en utilisant la console PYTHON :

```
>>> s="a,b,c,112,-3"
>>> L=s.split(",")
>>> print L
['a', 'b', 'c', '112', '-3']
>>> x=int(L[3])
>>> y=int(L[4])
>>> print x+y
109
```

L'aide de PYTHON pour la commande `split` :

```
string.split = split(s, sep=None, maxsplit=-1)
split(s [,sep [,maxsplit]]) -> list of strings
```

Return a list of the words in the string `s`, using `sep` as the delimiter string. If `maxsplit` is given, splits at no more than `maxsplit` places (resulting in at most `maxsplit+1` words). If `sep` is not specified or is `None`, any whitespace string is a separator.

(`split` and `splitfields` are synonymous)

Corrections

Q.1:

```
def moyenne(L):  
    s=0  
    for x in L:  
        s=s+x  
    return float(s)/len(L)
```

Temps d'exécution en $O(n)$.

Q.2: Le résultat renvoyé est le plus grand nombre apparaissant dans **L**. Le temps d'exécution est en $O(n)$. Invariant de boucle :

$$m = \max\{L[0], \dots, L[i-1]\}$$

Q.3:

```
def imaximum(L):  
    m=0  
    i=1  
    while i<len(L):  
        if L[i]>L[m]:  
            m=i  
        i=i+1  
    return m
```

Q.4:

```
T=[]  
f=open('/home/ba/Enseignement/PC/Python/Data/rfxcmd.txt','r')  
# On parcourt toutes les lignes du fichier f :  
for s in f.readlines():  
    # On décompose la chaîne de caractères  
    L=s.split(",")  
    T.append(float(L[3]))  
f.close()  
print T[:10]
```

[20.7, 20.7, 20.7, 20.7, 20.5, 20.7, 21.0, 20.7, 21.2, 20.6]

Q.5:

```
print moyenne(T)
```

18.614893617

Q.6:

```

T=[]
f=open('/home/ba/Enseignement/PC/Python/Data/rfxcmd.txt','r')
# On parcourt toutes les lignes du fichier f :
for s in f.readlines():
    # On décompose la chaîne de caractères
    L=s.split(",")
    if L[2]=="B904":
        T.append(float(L[3]))
f.close()
print moyenne(T)

```

20.8161290323

Q.7:

```

T1=[] # Températures de B904
T2=[] # Températures de FE02
f=open('/home/ba/Enseignement/PC/Python/Data/rfxcmd.txt','r')
# On parcourt toutes les lignes du fichier f :
for s in f.readlines():
    # On décompose la chaîne de caractères
    L=s.split(",")
    if L[2]=="B904":
        T1.append(float(L[3]))
    elif L[2]=="FE02":
        T2.append(float(L[3]))
f.close()
f=open('/home/ba/Enseignement/PC/Python/Data/rfxmoy.txt','w')
f.writelines(["B904,%s\n" % moyenne(T1),"FE02,%s\n" % moyenne(T2)])
f.close()

```

Q.8: Table **rfxcmd** (id :Int, time :Int, Ref :TEXT, Temp :FLOAT).

Q.9:

```
SELECT AVG(Temp) FROM rfxcmd
```

```
SELECT AVG(Temp) FROM rfxcmd WHERE Ref='B904'
```