INFORMATIQUE : DS 2

24 AVRIL 2017

EXERCICE 1 - AUTOUR DU MINIMUM

Dans toute cet exercice, on s'interdit l'usage de la fonction `min` préprogrammée en Python et permettant d'obtenir le minimum d'une liste donnée en argument. En revanche on donne la fonction `mini` suivante, écrite en Python.

```
def mini(t):
    ''' Calcule le minimum d'un tableau d'entiers ou de flottants'''
    if len(t) == 0:
        return None
    p = t[0]
    for i in range(len(t)):
        if t[i] <= p :
            p = t[i]
    return p
```

1. Expliquer le déroulement pas à pas lors de l'appel `mini([6,2,15,2,15])`, puis donner la valeur renvoyée.
2. Évaluer la complexité en comparaisons de l'appel `mini(t)` en fonction du nombre n d'éléments de `t`.
3. Proposer une fonction `pos_min(t)` dont la valeur renvoyée soit la (une) position où le minimum est atteint.
4. Préciser dans vote fonction `pos_min(t)` quelle position est renvoyée si le minimum est présent plusieurs fois dans la liste, et un moyen de changer ce comportement.

On souhaite maintenant déterminer la valeur minimale d'un tableau bidimensionnel d'entiers/de flottants. Un appel pourrait être :

```
>>> mini2D([[10,3,15],[5,13,10]])
>>> 3
```

5. Programmer une fonction réalisant `mini2d(t)` ce travail (on supposera le listes internes de taille non nulle).
6. Évaluer la complexité en comparaisons de cette fonction lorsque le tableau en entrée est de taille $n \times p$.

On voudrait maintenant, partant d'un liste constituée de couples (chaîne de caractère, entier) déterminer la/une chaîne de caractère pour laquelle l'entier associé est minimal :

```
>>> chaine_mini([['Bernard',40],['Bill',10],['Marcel',35]])
>>> 'Bill'
```

7. Programmer une fonction `chaine_mini(t)` réalisant cette opération.
8. Modifier ce programme pour qu'il renvoi toutes les chaines de caractères dont l'entier associé est minimal.

Afin d'élire le boulet d'or de l'année, les notes des élèves sont réparties dans un tableau :

```
Notes = [(Flantier,[10,8,15...],(Chambart,[15,11,13,...],...]
```

constitué de couples `(nom,liste)` dont le premier élément est le nom de l'élève et le second un tableau regroupant ses notes de colles de l'année. Le boulet d'or est celui dont la somme des notes est la plus basse.

9. Écrire un code en Python permettant d'afficher le nom du boulet d'or. On considère que la variable `Notes` est saisie dans le code.

EXERCICE 2 - LE TIPE DES VAINQUEURS

Ne pss chercher de vérités physiques derrière les énoncés. Les questions sont relativement indépendantes.
Pour son TIPE, Raoul étudie les variations d'intensité cérébrale d'un corps lâché du sommet d'une falaise. Ces variations sont données par une fonction $f(t)$ dépendant du temps t qu'il cherche à modéliser.

1. Ne sachant pas quoi faire de son modèle, il décide de rechercher quand la fonction f s'annule. Il pense utiliser la méthode Newton pour obtenir une valeur approchée à un `eps` près.
   Après avoir rappelé le principe de la Méthode de Newton sur un schéma, écrire une fonction `Newton(f,fp,eps,x0)` qui renvoie une valeur approchée d'un zéro de `f` à `eps` près en partant de `x0` en utilisant f' =`fp`.
2. Se rendant compte qu'il n'a pas accès à la dérivée f' de f, il décide d'approcher ces valeurs via une formule qu'il trouve sur internet : en x_0 on a $\frac{f(x_0+h)-f(x_0-h)}{2h} - f'(x_0) \sim \frac{f^{(3)}(x_0)}{6}h^2$. Montrez à l'aide de la formule de Taylor-Young de f à l'ordre 3 en $x+h$ et $x-h$ que cette formule est juste.

Pour chaque expérience il souhaite récupérer les données de cette fonction dans une variable globale `Lmes` des valeurs calculées tout les millièmes de secondes à partir de $t = 0$.

```
Lmes = [0.235 , 0.894, 1.564, ... ]
```

Chaque valeur est encodée sur 64 bits via la norme IEEE 754.

3. Il ne dispose qu'une d'une vielle clé USB de 1Mo. Sachant qu'il voudrait faire une dizaine d'expériences de 1 minute chacune, pourra-t-il stocker tous les résultats ?

Il se dit qu'une bonne idée serait de calculer l'intégrale de f sur la durée d'acquisition :

$$\text{int}_f = \int_0^{t_{tot}} f(t)dt$$

La valeur t_{tot} étant liée à la liste `Lmes`, dont on rappel que le pas est $0,001s$.

4. Écrire une suite d'instructions qui calcule cette valeur avec la méthode des trapèzes.

L'intégrité morale $m(t)$ d'un corps en chute libre au cours du temps t en secondes suit une équation différentielle

$$m'' + tm' - m^2 = 0$$
$$m(0) = m'(0) = 100$$

Raoul décide de résoudre cette équation différentielle via la méthode d'Euler.

5. Il regarde son cours et y trouve qu'il faut poser $Y = (m, m')$ pour obtenir une équation différentielle sous la forme $Y' = F(t, Y)$. Déterminer la fonction mathématique F modélisant cette équation différentielle en précisant ses caractéristiques.
6. Ne comprenant rien à son code de cours, il vous demande de le compléter :

```
def F(t,Y):
     """ Fonction definissant l'équation différentielle"""
    m,mp = Y
    return # à compléter

# Paramètres
tmax = 200
tmax = 60
Y0 = np.array([100,100])
h = 0.001
N = int(tmax/h)

t = 0
Y = X0
tt = [t]
YY = [Y]

# Euler
#for i in range(N):
for k in range(N) :
    Y = # à compléter
    t = # à compléter
    tt.append(t)
    YY.append(Y)
```

7. Pour compléter son TIPE Raoul veut construire le portrait de phases de m pour une durée d'une minute. Il se souvient qu'il doit commencer par écrire `import matplotlib.pyplot as pl` dans son code. Écrire le code permettant de réaliser le souhait de Raoul, en considérant que le code de la question précédente est déjà saisi.