INFORMATIQUE MP DEVOIR 2

12 JANVIER 2017

Le soin et la clarté, ainsi que le respect de la syntaxe en Python sont partie intégrante de la notation finale.

C'EST LA FIN DU MONDE !

Un plan extérieur tente d'envahir notre plan d'existence !
Afin d'endiguer la fin du monde, des scientifiques tentent de mesurer les distorsions spatio-temporelles dans divers secteurs.

Des détecteurs analogiques sont placés en des endroits stratégiques pour détecter les positrons et les tachyons présents.
Les positrons montrent une réaction de notre plan afin d'assurer la stabilité.
Les tachyons montrent une activité du plan extérieur pour forcer le passage.

Il est de votre devoir de Taupins de les aider.

PARTIE A- RÉCUPÉRATION DES DONNÉES

Chaque détecteur relève les présences de tachyons et de positrons et converti le signal analogique en entier. Les valeurs entières sont positives ou négatives, et ne dépassent pas 100 en valeur absolue.

1. Sur combien de bits doit-on encoder une valeur mesurée ? On supposera que le codage des entiers est fait en complément à 2.
2. Une mesure complète lors d'une heure (mesure + informations) prends 1 Mo. On dispose les stations sur une surface de 100m × 80m à raison d'une station par m^2. Si on fait des mesures pendant 12h, de quelle quantité de mémoire doit-on disposer sur le serveur ?

Une mesure complète est contenue dans un tableau Python `mesures` dont un exemple est le suivant :

```
mesures = [('P',[23,31]),('T',[-24]),('P',[-3]),('T',[12,53,-50]) ,('C',42)]
```

Un tableau `mesures` contient des couples :
- La première valeur du couple est un caractère correspondant au type de données : 'T' pour tachyons, 'P' pour positron, 'C' pour le dernier couple correspondant à un checksum.
- Ensuite pour les valeurs mesurées, le deuxième élément du couple est un tableau des valeurs détectées. Attention les tableaux n'ont pas nécessairement la même taille.
- Pour le dernier couple, le deuxième élément est un checksum correspondant à la somme des valeurs mesurées dans les autres couples.

On veut créer une fonction en Python `repart` qui renvoie un triplet `(P,T,chk)` tel que `P` et `T` soient des tableaux contenant les valeurs mesurées, et `chk` la valeur du checksum.

```
>>> repart(mesures)
([23, 31, -3], [-24, 12, 53, -50], 42)
```

3. Écrire cette fonction `repart`.

Pour s'assurer de la justesse de la transmition, il est important de vérifier que le checksum est bon, c'est à dire qu'il est égal à la somme de toutes les valeurs mesurées.

```
>>> checksum(mesures)
True
```

4. Écrire cette fonction `checksum` qui renvoie un booléen selon que le checksum soit bon ou pas.

PARTIE B - COHÉRENCE DES DONNÉES

On suppose que les valeurs mesurées sont saisies dans deux tableaux `P` et `T`.
Comme les données à observer sont extrêmement difficiles à mesurer il faut s'assurer de leur cohérence.
D'après le professeur Xavier une mesure est cohérente pour un type de données si l'écart entre la moyenne des

valeurs absolues mesurées et le double de la valeur moyenne obtenue par la méthode des trapèzes est inférieur à 1. Autrement dit si :

$$|Moy_{abs} - \frac{2}{\Delta t}\int_{t_1}^{t_2} f(t)dt| < 1$$

Pour chaque tableau l'intervalle de temps est simplement sa longueur.

1. Écrire une fonction `somme_abs(t)` qui renvoie la moyenne des valeurs absolues d'un tableau t.
2. Écrire une fonction `trapeze(t)` qui renvoie la valeur moyenne d'une intégrale par la méthode des trapèzes pour des valeurs contenues dans t (avec un pas de 1).
3. Ecrire une fonction `coherent(P,T)` qui renvoie un booléen selon que les deux mesures soit cohérentes ou non.

PARTIE C - BASE DE DONNÉES

Afin de gérer les données reçues, on a construit une base de données contenant deux tables dont des exemples sont écrits ci-dessous.

scentifique		
iden	nom	centre
001AD	Alpha	sud-ouest
121BE	Beta	nord-est
011DE	Gamma	nord-est

mesure			
id	idnom	type	fichier
000001	001AD	detection	mes0021.csv
000002	121BE	test	test001.csv
000321	001AD	detection	mes0035.csv

La table `scientifique` contient les colonnes :
- iden : un identifiant unique du scientifique
- nom : le nom (pseudonyme pour la sécurité) du scientifique
- centre : le centre sur lequel est basé le scientifique

La table `mesure` contient les colonnes :
- id : un identifiant unique de la mesure
- idnom : le numéro d'identification du scientifique
- type : le type de mesure effectuées
- fichier : le nom du fichier correspondant à la mesure

1. Écrire une requête SQL donnant les noms des scientifiques du centre nord-est.
2. Écrire une requête SQL renvoyant le nom du scientifique, le type de mesure et le fichier correspondant pour chaque mesure.
3. Écrire une requête SQL renvoyant le nombre de mesures de chaque type effectuées.
4. Écrire une requête SQL donnant le nom du scientifique ayant fait le plus de mesures de type détection, avec ce nombre.

PARTIE D - CONSTRUCTION D'UNE IMAGE

Le but est maintenant de construire une image en noir et blanc pour détecter les zones de risques : le noir est une risque fort et le blanc un risque faible.
Dans une zone de 100m × 80m on été placés un détecteur par m^2.
Les relevés sont situés dans un tableau `releves` contenant des quadruplet `(x,y,P,T)` où :
- `x` et `y` sont les coordonnées du détecteur, avec $0 \leqslant x \leqslant 100$ et$0 \leqslant x \leqslant 80$
- `P` et `T` sont les valeurs relevées des positrons et des tachyons par le détecteur.

Une image en niveau de gris est un tableau de tableau `image` en Python. Pour simplifier la valeur `image[y][x]` de chaque pixel de l'image doit contenir un niveau de gris (entre 0 pour noir et 1 pour blanc) pour la station de coordonnées `(x,y)`.
Une image en noir et blanc suit le même principe mais ne prends comme valeurs que des 0 ou des 1.
Le calcul du niveau de gris N se fait par $N = \frac{|P|}{|T|+|P|}$.
Le niveau de bascule entre le niveau de gris et le noir ou blanc est de 0,75.

1. Écrire une fonction `niveau(P,T)` qui renvoie la valeur `N` du niveau de gris selon se calcul.
2. Écrire une fonction `image_gris(releve)` qui construit et renvoie cette image en niveau de gris.
3. Écrire une fonction `image_risques(releve)` qui construit et renvoie l'image des risques en noir et blanc.