

Méthodologie de la Programmation

TD - 15 Septembre 2017

Horaires de trains

Représentation du temps

Différentes représentations du temps peuvent coexister au sein d'un programme. Nous allons travailler ici avec la représentation textuelle d'un *datetime* au format ISO, sous la forme simplifiée suivante : 'AAAA-MM-JJ HH:MM:SS'; par exemple "2017-09-15 10:00:00" pour le 15 septembre 2017 à 10h00.

Objectif

Soit un fichier texte "trains.txt" contenant les horaires de départ et d'arrivée de trains, au format suivant:

```
TRAIN 1:
datetime départ
datetime arrivée
TRAIN 2:
datetime départ
datetime arrivée
...
TRAIN n:
datetime départ
datetime arrivée
```

On veut calculer la durée de chaque voyage, et enregistrer les résultats dans un second fichier.

Attention, les jours de départ et d'arrivée peuvent différer. En revanche, pour des raisons de simplicité, on considère que le mois sera toujours le même.

Les exercices suivants devraient être réalisés dans l'ordre.

Merci de rédiger dans un fichier .py portant votre nom, prénom et n. d'étudiant.

Vous devez envoyer votre travail (script + fichier de sortie) sur le mail

josue.melka02@univ-paris8.fr à la fin de l'épreuve. Tout travail rendu après le délai fixé ne pourra être pris en compte.

Exercice 1: (5 points)

Ecrire une fonction `iso_to_tuple` qui transforme un *datetime* au format ISO en tuple d'entiers. Cette fonction aura pour signature:

```
def iso_to_tuple(dt):  
    ...
```

Par exemple :

```
>>> iso_to_tuple("2017-09-15 10:00:00")  
(2017, 9, 15, 10, 0, 0)
```

Exercice 2: (5 points)

On veut connaître l'écart en secondes entre deux *datetimes* donnés au format texte. *Pour simplifier, on considère que les deux datetimes appartiennent toujours au même mois et la même année.*

Écrire une fonction qui retourne la différence en secondes entre ses deux arguments. Le résultat correspond à $(dt1 - dt2)$, et peut donc être négatif. Cette fonction aura pour signature:

```
def calc_diff(dt1, dt2):  
    ...
```

Par exemple :

```
>>> calc_diff("2012-10-02 20:10:00", "2012-10-05 15:20:00")  
-241800
```

Exercice 3: (10 points)

Ecrire un script qui lit le fichier "trains.txt", calcule la durée de chaque voyage en heures/minutes, et enregistre le résultat dans le fichier "durees.txt". Le format de sortie attendu est le suivant (en cas de difficulté, restez au format le plus proche possible):

```
TRAIN 1: hh:mm  
TRAIN 2: hh:mm  
...  
TRAIN n: hh:mm
```