

Cette situation d'évaluation comporte ce document ainsi que des fichiers de codes et de données présents sur l'ordinateur à la disposition du candidat. Le candidat doit restituer ce document avant de sortir de la salle d'examen. Le candidat doit agir en autonomie et faire preuve d'initiative tout au long de l'épreuve.

En cas de difficulté, le candidat peut solliciter l'examineur afin de lui permettre de continuer la tâche. Des moments privilégiés pour solliciter l'examineur sont indiqués dans le document sous la forme d'appels professeur.

L'examineur peut intervenir à tout moment, s'il le juge utile.

Les compteurs d'eau modernes permettent de mesurer automatiquement la consommation d'un foyer. Ils enregistrent, chaque heure, plusieurs informations.

Les données sont fournies sous forme d'une liste de dictionnaires. On considère que cette liste est triée par jour et heure croissant. Chaque mesure est un dictionnaire contenant :

- "jour" : le jour de la mesure (chaîne de caractères au format "AAAA-MM-JJ") ;
- "heure" : l'heure de la mesure (chaîne de caractères au format "HH:MM") ;
- "chaude" : le volume d'eau chaude consommé en litres depuis la mesure précédente (entier) ;
- "froide" : le volume d'eau froide consommé en litres depuis la mesure précédente (entier).

Exemple de mesure :

```
{"jour": "2025-02-04", "heure": "08:00", "chaude": 5, "froide": 8}
```

La consommation totale d'une mesure est la somme de l'eau chaude et de l'eau froide.

L'objectif de ce sujet est d'écrire plusieurs fonctions manipulant ces données, puis d'analyser et de corriger une fonction existante qui contient une erreur.

Question 1

Écrire une fonction `total_conso` qui prend en paramètres : - `donnees` : une liste de mesures - `jour` : une chaîne représentant le jour (ex. "2025-02-04") et renvoie la consommation totale d'eau (somme de l'eau chaude et de l'eau froide) de toutes les mesures pour ce jour.

Par convention, si aucune mesure n'existe pour ce jour, la fonction renvoie `None`.

Par exemple :

```
>>> total_conso(donnees, "2025-02-04")
33
>>> total_conso(donnees, "2025-12-25")
>>>
```



Appeler le professeur pour lui présenter votre réponse ou en cas de difficulté.

Question 2

On considère que la nuit, quand tout le monde dort, la consommation d'eau n'est pas censée être supérieure à zéro pendant plusieurs heures consécutives. Une fuite est donc suspectée lorsqu'il y a au moins 3 mesures consécutives entre 00:00 et 05:00 inclus où la consommation totale est toujours non nulle (eau chaude + eau froide > 0).

Écrire une fonction `fuite_possible` qui renvoie `True` si une fuite est possible ce jour-là, `False` sinon. Cette fonction prend en paramètres :

- `donnees` : une liste de mesures ;
- `jour` : une chaîne de caractères représentant la date, au format "AAAA-MM-JJ".

Cet exemple renverrait `False`, car il n'y a pas trois mesures consécutives non nulles :

Heure	00:00	01:00	02:00	03:00	04:00	05:00
Consommation	5	3	0	3	0	0

Cet exemple renverrait `True`, car il y a trois mesures consécutives non nulles (00:00, 01:00, 02:00) :

Heure	00:00	01:00	02:00	03:00	04:00	05:00
Consommation	2	1	1	0	0	0



Appeler le professeur pour lui présenter votre réponse ou en cas de difficulté.

Question 3

Le fichier `analyse_eau.py` contient une fonction appelée `lissage_conso` censée calculer une moyenne sur 3 valeurs pour lisser les mesures de la consommation.

Pour les cas particuliers :

- **premier élément** faire la moyenne du premier et du deuxième ;
- **dernier élément** faire la moyenne du dernier et de l'avant-dernier ;
- **éléments intermédiaires** faire la moyenne de trois valeurs (précédente, actuelle, suivante).

La fonction doit toujours renvoyer une liste de même taille que la liste d'origine.

Pour chaque valeur, on fait la moyenne avec ses voisins (précédent et suivant). Cependant, cette fonction contient une erreur.

Expliquer pourquoi la fonction `lissage_conso`, testée avec la liste suivante, présente un résultat incorrect, et proposer une correction.

```
test = [10, 20, 30, 40, 50]
```



Appeler le professeur pour lui présenter votre réponse ou en cas de difficulté.

Question 4

La fonction `lissage_conso` prend en compte les cas limites dans lesquels la liste fournie contient seulement deux valeurs. Identifier un autre cas limite qui n'est pas pris en compte par la fonction, et proposer une solution pour y remédier.



Appeler le professeur pour lui présenter votre réponse ou en cas de difficulté.

Description du dossier

Le dossier fourni au candidat sur l'ordinateur comporte les éléments suivants :

- une version PDF de l'énoncé ;
- `analyse_eau.py` : fonction à corriger + squelette des autres fonctions.