

BACCALAURÉAT

SESSION 2026

Épreuve de l'enseignement de spécialité

NUMÉRIQUE et SCIENCES INFORMATIQUES

Partie pratique

Classe Terminale de la voie générale

Sujet n°20

DURÉE DE L'ÉPREUVE : 1 heure

**Le sujet comporte 4 pages numérotées de 1 / 4 à 4 / 4
Dès que le sujet vous est remis, assurez-vous qu'il est complet.**

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.

Le numérique représente aujourd'hui environ 4 % des émissions mondiales de gaz à effet de serre, soit davantage que le transport aérien civil. Une part importante de cette empreinte provient de la fabrication des équipements, mais l'usage quotidien (streaming vidéo, envoi d'emails, stockage cloud) y contribue également de manière significative.

On souhaite développer une application permettant d'estimer l'empreinte carbone d'un utilisateur en fonction de ses usages numériques. Les émissions sont exprimées en gramme de CO₂ équivalent (gCO₂e).

On dispose des données suivantes concernant les émissions moyennes de différentes activités numériques :

Activité	Émission unitaire	Unité
Email sans pièce jointe	4	gCO ₂ e par email
Email avec pièce jointe (1 Mo)	19	gCO ₂ e par email
Streaming vidéo SD	36	gCO ₂ e par heure
Streaming vidéo HD	100	gCO ₂ e par heure
Recherche web	7	gCO ₂ e par recherche
Stockage cloud	10	gCO ₂ e par Go par mois

Un utilisateur est représenté par un dictionnaire contenant ses usages mensuels. Par exemple :

```
utilisateur1 = {
    'emails_simples': 150,
    'emails_pj': 20,
    'streaming_sd': 10, # en heures
    'streaming_hd': 25, # en heures
    'recherches': 500,
    'stockage_cloud': 15 # en Go
}
```

Question 1

Dans le fichier `code_empreinte.py`, écrire en Python le code de la fonction `calculer_empreinte(utilisateur)` qui prend en paramètre un dictionnaire représentant les usages

d'un utilisateur (au format décrit précédemment) et qui renvoie l'empreinte carbone totale mensuelle en gCO₂e qui est de type entier.
Par exemple, `calculer_empreinte(utilisateur1)` renvoie la valeur 7490.



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

Question 2

Écrire en Python le code de la fonction `classer_par_impact(utilisateur)` qui prend en paramètre un dictionnaire représentant les usages d'un utilisateur et qui renvoie un dictionnaire contenant trois clés 'fort', 'moyen' et 'faible', chacune associée à une liste des noms des activités correspondantes selon les critères suivants :

- impact 'fort' : émissions ≥ 1000 gCO₂e
- impact 'moyen' : $200 \text{ gCO}_2\text{e} \leq \text{émissions} < 1000 \text{ gCO}_2\text{e}$
- impact 'faible' : émissions < 200 gCO₂e

Par exemple, pour l'utilisateur `utilisateur2` ayant :

- `streaming_hd` : 15 (1500 gCO₂e) → impact 'fort'
- `emails_simples` : 100 (400 gCO₂e) → impact 'moyen'
- `recherches` : 10 (70 gCO₂e) → impact 'faible'

La fonction doit renvoyer dans ce cas :

```
{
  'fort': ['streaming_hd'],
  'moyen': ['emails_simples'],
  'faible': ['recherches']
}
```



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

La fonction `comparer(utilisateur1, utilisateur2)`, dont le code est fourni, prend en paramètres deux dictionnaires `utilisateur1` et `utilisateur2` représentant les usages de deux utilisateurs. Elle renvoie un dictionnaire associant à chaque activité la différence des émissions entre les deux utilisateurs. Cette différence est définie comme l'émission de l'utilisateur 2 moins celle de l'utilisateur 1. Si une activité est absente chez un utilisateur, on considère que son émission vaut 0.

Question 3

Compléter le code de la fonction `test_comparer` en ajoutant au moins deux tests pertinents qui permettent de vérifier différents aspects du comportement de la fonction. Pour chaque test ajouté, une brève justification doit être donnée afin d'expliquer pourquoi ce cas est

important à vérifier. D'autres utilisateurs peuvent être créés lors de l'écriture des tests.

Pour mieux percevoir l'écart entre les usages de deux utilisateurs, on utilise la formule ci-dessous :

$$\text{pourcentage d'écart} = \frac{\text{émission2} - \text{émission1}}{\text{émission1}} \times 100$$

La fonction `comparer_v2(utilisateur1, utilisateur2)`, dont le code est fourni, prend en paramètres deux dictionnaires représentant les usages de deux utilisateurs. Elle renvoie un dictionnaire associant à chaque activité le pourcentage d'écart expliqué ci-dessus.

Par exemple, `comparer_v2(utilisateur1, utilisateur2)` renvoie :

```
{
  'emails_pj': -100.0,
  'emails_simples': -33.33333333333333,
  'recherches': -98.0,
  'stockage_cloud': -100.0,
  'streaming_hd': -40.0,
  'streaming_sd': -100.0
}
```

Dans certains cas, on constate que cette fonction provoque une erreur.

Question 4

Identifier dans quels cas l'erreur se produit et proposer une correction.



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 ;
- un code source de départ `code_empreinte.py`