

**BACCALAURÉAT
SESSION 2026**

**Épreuve de l'enseignement de spécialité
NUMÉRIQUE et SCIENCES INFORMATIQUES**

Partie pratique

Classe Terminale de la voie générale

Sujet n°13 — Variante d'entraînement

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.

On considère dans ce sujet un fichier CSV de données océanographiques collectées par une **bouée dérivante instrumentée**. Larguée en mer depuis un navire de recherche, cette bouée dérive librement sous l'effet des courants marins tout en descendant progressivement grâce à un système de ballast. Elle est équipée de capteurs qui mesurent la **pression en pascals (Pa)**, la **salinité en grammes par kilogramme (g/kg)**, l'**altitude négative (profondeur) en mètres** et sa **géolocalisation GPS**. Ces données sont essentielles pour l'étude des masses d'eau océaniques, des courants de fond et du climat. Pour simplifier l'étude, on ne dispose que de données mesurées lors de la descente.

Source : Ifremer / Argo Float Program

Ce sujet propose de concevoir un script Python qui exploite les données océanographiques relevées avec :

- la conversion des valeurs de pression dans une unité plus courante ;
- une recherche d'indice de salinité maximale et de profondeur(s) correspondante(s) ;
- la génération d'un fichier d'extension GPX (GPS Exchange Format) compatible avec de nombreuses applications de cartographie.

On donne une première fonction Python `recupere_donnees_fichier_csv(nom_fichier)` qui permet d'ouvrir le fichier CSV, de supprimer les en-têtes et de récupérer les données relevées sous forme de 5 listes.

Question 1

En utilisant la fonction `recupere_donnees_fichier_csv`, écrire la ou les lignes de code qui récupèrent les données relevées par la bouée océanique dans 5 listes différentes (profondeurs, pressions, salinités, longitudes et latitudes).

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

La pression en pascals (Pa) est liée à la pression en bar selon la relation : **1 bar = 100 000 Pa**. La pression en bar s'obtient donc en divisant la pression en pascals par 100 000.

Question 2

Écrire en Python une fonction nommée `conversion_Pa_en_bar` qui prend en paramètre une liste de pressions en pascals (Pa) et retourne cette même liste de pressions, mais converties en bars (bar). Les valeurs doivent être arrondies à 4 chiffres après la virgule (commande utilisable : `round(valeur, nb_chiffres)`). Écrire une ligne de code permettant de tester la fonction proposée.

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

On souhaite disposer d'une fonction nommée `profondeur_la_plus_salee` qui prend en paramètres une liste de profondeurs ainsi qu'une liste de salinités en g/kg. Cette fonction doit renvoyer la salinité maximale, ainsi qu'une liste contenant la ou les profondeurs correspondantes.

Par exemple, avec les deux listes suivantes :

Exemple 1 :

```
>>> profondeurs = [500, 1000, 2500, 4000, 4000]
>>> salinites = [34.99, 35.25, 36.44, 37.74, 37.74]
>>> profondeur_la_plus_salee(profondeurs, salinites)
(37.74, [4000, 4000])
```

Exemple 2 :

```
>>> profondeurs = [200, 750, 1500, 3000, 6000]
>>> salinites = [34.66, 35.08, 35.59, 36.86, 39.27]
>>> profondeur_la_plus_salee(profondeurs, salinites)
(39.27, [6000])
```

Question 3

Proposer une écriture de la fonction `profondeur_la_plus_salee`.

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

On donne une deuxième fonction Python `genere_gpx` qui prend en paramètres 2 listes (`liste_longitudes` et `liste_latitudes`), données de géolocalisation de la bouée, et les utilise pour générer un fichier d'extension GPX. Ce format normalisé utilise des balises XML ouvrantes et fermantes.

Question 4

Observer le corps de la fonction `genere_gpx`, puis, à l'aide d'une assertion, insérer une ligne de code qui garantit que les deux listes `liste_longitudes` et `liste_latitudes` sont de même longueur.

Question 5

Écrire une ligne de code qui appelle la fonction `genere_gpx` avec les deux listes de longitudes et latitudes obtenues à la question 1, puis ouvrir le fichier GPX généré dans le même répertoire que le fichier Python.

Après analyse, il s'avère que ce fichier GPX généré pose des problèmes de compatibilité avec certains logiciels de cartographie. En effet, la balise racine GPX n'a pas été fermée en toute fin de fichier (`</gpx>` absente).

Question 6

Proposer une amélioration du code visant à répondre à cette difficulté.

■ 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 `etude_oceanographique.py` ;
- un fichier des données océanographiques au format CSV nommé `releves_bouee_ocean.csv`.