

# **BACCALAURÉAT**

**SESSION 2026**

---

**Épreuve de l'enseignement de spécialité**

## **NUMÉRIQUE et SCIENCES INFORMATIQUES**

**Partie pratique**

**Classe Terminale de la voie générale**

---

**Sujet n°13**

---

**DURÉE DE L'ÉPREUVE : 1 heure**

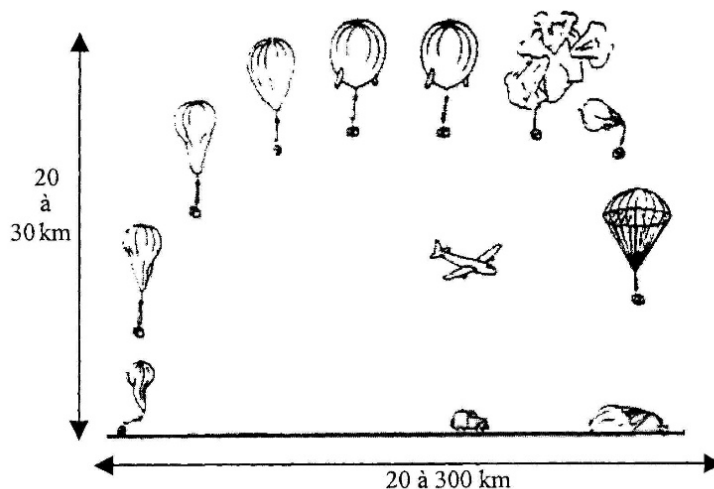
**Le sujet comporte 4 pages numérotées de 1 / 4 à 4 / 4  
Dès que le sujet vous est remis, assurez-vous qu'il est complet.**

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 climatiques (`releve_ballon_sonde.csv`) de Météo France obtenues par ballon sonde. Il est gonflé à l'aide d'un gaz léger (hélium) et est équipé d'instruments pour acquérir les paramètres tels que la température en kelvins, l'altitude en mètres et sa géolocalisation. Lorsqu'il est lâché, ce ballon sonde s'élève et collecte des données tout au long de sa trajectoire, puis éclate à une altitude appropriée ce qui entraîne sa chute au sol avec un parachute pour amortir sa vitesse de descente. Ces données relevées sont essentielles pour les prévisions météorologiques car elles fournissent des informations sur les conditions atmosphériques à différentes altitudes. Pour simplifier l'étude, on ne dispose que de données mesurées durant la montée.



Source : CNES-Planète sciences

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

- la conversion des valeurs de températures dans une unité plus courante ;
- une recherche de basse température et d'altitude(s) correspondante(s) ;
- la génération d'un fichier d'extension `km1` (Keyhole Markup Language) 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 4 listes.

### Question 1

En utilisant cette fonction `recupere_donnees_fichier_csv`, écrire la ou les lignes de code qui récupèrent les données relevées par le ballon sonde dans 4 listes différentes (`altitudes`, `temperatures`, `longitudes` et `latitudes`).



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

La température en kelvins correspond à la température en degrés celsius à laquelle on ajoute la valeur 273,15.

### Question 2

Écrire en Python une fonction nommée `conversion_K_en_C` qui prend en paramètre une liste de températures en kelvins (K) et retourne cette même liste de températures, mais converties en degrés Celsius (°C). Les valeurs de températures doivent être arrondies à 1 chiffre après la virgule (commande utilisable : `round(valeur, nb chiffres après la virgule)`). 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 `altitude_la_plus_froide` qui prend en paramètres une liste d'altitudes, ainsi qu'une liste des températures en degrés Celsius. Cette fonction doit renvoyer la température la plus froide, ainsi qu'une liste contenant la ou les altitudes correspondantes.

Par exemple, avec les deux listes `altitudes` et `temperatures` suivantes :

Exemple 1 :

```
>>> altitudes = [7000, 10125, 13896, 14211]
>>> temperatures = [-35.2, -52.1, -57.4, -57.4]
>>> altitude_la_plus_froide(altitudes,temperatures)
(-57.4, [13896, 14211])
```

Exemple 2 :

```
>>> altitudes = [6000, 7250, 11542, 15214, 17300]
>>> temperatures = [-33.7, -45, -53, -58.5, -60.1]
>>> altitude_la_plus_froide(altitudes,temperatures)
(-60.1, [17300])
```

### Question 3

Proposer une écriture de la fonction `altitude_la_plus_froide`.



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

On donne une deuxième fonction Python `genere_kml` qui prend en paramètres 2 listes (`liste_longitudes` et `liste_latitudes`), qui sont les données relevées de géolocalisation du ballon sonde, et les utilise pour générer un fichier d'extension `kml`. Ce format normalisé `kml` utilise des balises ouvrantes et fermantes.

#### Question 4

Observer le corps de la fonction `genere_kml`, 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_kml` avec les deux listes de longitudes et latitudes obtenues à la question 1, puis ouvrir le fichier `kml` généré dans le même répertoire que le fichier Python.

Après analyse, il s'avère que ce fichier `kml` généré pose des problèmes de compatibilité avec certains logiciels de cartographie. En effet, la balise `kml` n'a pas été fermée en toute fin de fichier (`</kml>` 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_climatique.py` ;
- Un fichier des données météorologiques au format `csv` nommé `releves_balloon_sonde.csv`.