

**PARTIE 1: INTERPRÉTATION DE LA TRAME D'UN GPS DE TÉLÉPHONE****> 1. PRÉAMBULE**

Le but de cette partie est d'apprendre à interpréter les données géographiques à partir d'un signal GPS reçu (appelé trame GPGGA) par un appareil situé au sol (smartphone, boîtier situé dans une voiture,...)

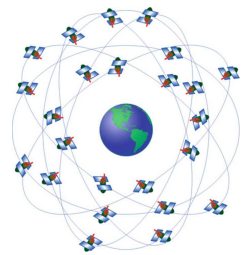
**? 2. EXEMPLE**

Un récepteur GPS reçoit le type de signal suivant :

**\$GPGGA,064036.289,4836.5375,N,00740.9373,E,1,04,3.2,200.2,M,,,\*0E**

Comment interpréter ce signal ?

\$GPGGA	Type de trame (ici trame GPS)
064036.289	Trame envoyée à 06h 40m 36,289s (heure UTC)
4836.5375,N	Latitude : $48^{\circ}36,5375' = 48,608958^{\circ}$ Nord
00740.9373,E	Longitude : $7^{\circ}40.9373' = 7,682288^{\circ}$ Est
1	Type de positionnement ( 1 signifie positionnement GPS)
04	Nombre de satellites utilisés pour calculer les coordonnées
3.2	Précision horizontale ou HDOP (Horizontal dilution of precision)
200.2,M	Altitude 200,2 mètres

**E 3. EXERCICE**

1. Votre récepteur GPS capte la trame suivante :

**\$GPGGA,113239.512,4545.47208,N,0449.93164,E,1,08,0.6,3.4,155.3,M,,,\*4A"**

a) Analyser cette trame pour déterminer les paramètres ci-dessous :

- latitude :
- longitude :
- latitude :
- heure d'émission de la trame :
- nombre de satellites utilisés :

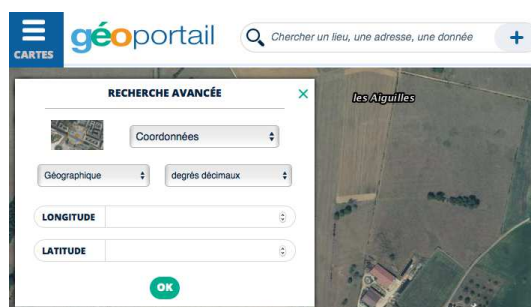
2. a) Ouvrir un navigateur et aller sur <https://www.geoportail.gouv.fr>

b) Cliquer "rechercher un lieu" puis, dans le champ de recherche, cliquer sur **+** pour accéder à la recherche avancée.

c) Faire défiler le menu déroulant sur "**Coordonnées**".

d) Saisir les coordonnées précédentes afin de découvrir votre position.

e) Où se situe votre capteur GPS ? : .....



**> 1. BUT DE CETTE PARTIE**

Dans cette partie, nous allons coder le travail précédent en Python

**? 2. EXERCICE**

1. Lancer Thonny et créer/sauvegarder un nouveau programme appelé **trameGPS.py**
2. Créer un chaîne de caractère regroupant la trame reçue par le boîtier GPS :

```
trame = "$GPGGA,113239.512,4545.47208,N,0449.93164,E,1,08,0.6,3.4,155.3,M,,,*4A"
```

3. Créer une liste en découpant la trame à chaque virgule :

```
liste = trame.split(',')
```

4. Désormais liste[0] contient **\$GPGGA**, liste[1] contient **113239.512**, liste[2] contient **4545.47208**, etc.

- a) Saisir la fonction suivante afin de déterminer la latitude d'une liste GPS

```
def latitude(liste):
    valeur = float(liste[2])           #il s'agit dans l'exemple de 4545.47208
    degre = int(valeur/100)           #chiffres dépassant les centaines (ici : 45)
    minutes = valeur%100              #chiffres jusqu'aux centaines (ici : 45.47208)
    if liste[3]=="N":                 #on est au Nord
        return degre + minutes/60     #la latitude est positive
    else :
        return -(degre +minutes/60)   #la latitude est negative
```

- b) Exécuter ce programme puis le tester en rajoutant, au choix :

— soit dans le programme, la ligne :

```
print(latitude(liste))
```

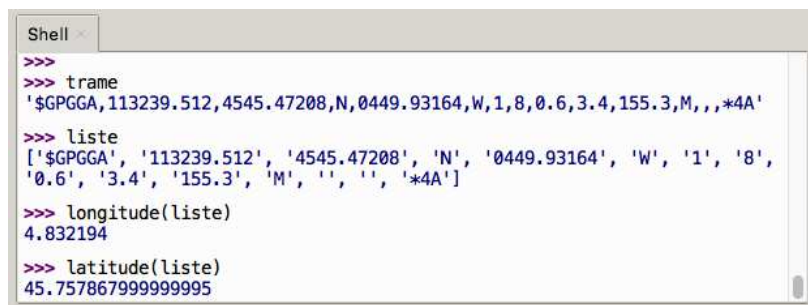
— soit dans le Shell, la ligne :

```
>>> latitude(liste)
```

- c) Effectuer le même travail pour la longitude :

```
def longitude(liste):
    # fonction à compléter
```

5.



- a) À partir de votre programme, déterminer et afficher les coordonnées GPS relatives à la trame suivante relevée par un boîtier GPS :

**\$GPGGA,071512.34,4851.1791,N,0220.9959,E,1,04,0.6,3.4,62.3,M,,0000\*0B**

- b) Où se situe l'appareil ?

**DEPÔT 1** trameGPS.py sur <http://entraide-ella.fr>

6. Un ami vous envoie une photo prise depuis son téléphone.

Sur le « TAG » de cette photo on peut lire : **latitude : 45,832425°** et **longitude : 6,864404°**

- a) D'où a été prise sa photo ? : .....
- b) À quelle altitude se situe votre ami ? : .....

