



FICHE DE TRAVAIL

Activité 1

GPS

Exercice 1 : Aller sur le site <https://www.openstreetmap.org/>

- 1) Manipuler la carte pour trouver votre lycée.
- 2) A l'aide de l'URL fournie par le navigateur donner les coordonnées GPS de la ville d'Arras.

2) Quelle est la ville qui se situe aux coordonnées $22^{\circ} 54' 39.6''$ S et $43^{\circ} 12' 7''$ O
Ecrire plutôt les coordonnées sous forme de deux nombres décimaux : -22.911, -43.2019

Exercice 2 :

Compléter les coordonnées GPS de la ville de Berlin par N ou S et E ou O.

$52^{\circ} 31' 1.3''$ Et $13^{\circ} 23' 19.9''$

Exercice 3 :

Aller sur le site OpenStreetMap.org et coller les coordonnées GPS :

40.741895,-73.989308 (coordonnées donnés sous forme de deux nombres décimaux)

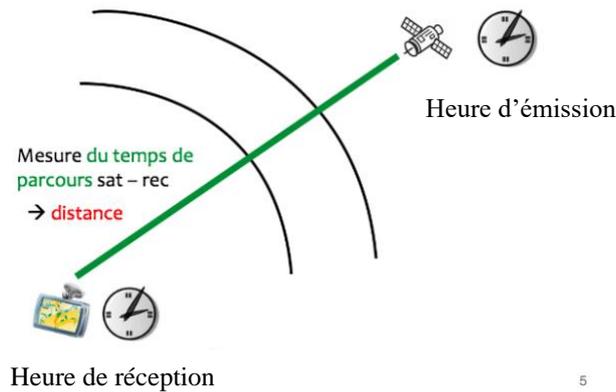
1) De quelle ville sont-ils les coordonnées ?

2) En cliquant sur la « Requête des objets » (la flèche avec un point d'interrogation, à droite), vous pouvez avoir des informations sur les bâtiments.
Cliquer sur le bâtiment qui se trouve sur East 26th Street.

Que représente ce bâtiment ?

Exercice 4 :

Pour déterminer la distance à laquelle un récepteur GPS se trouve d'un satellite, il repère l'heure à laquelle il reçoit un message et l'heure à laquelle le message a été envoyé par le satellite.



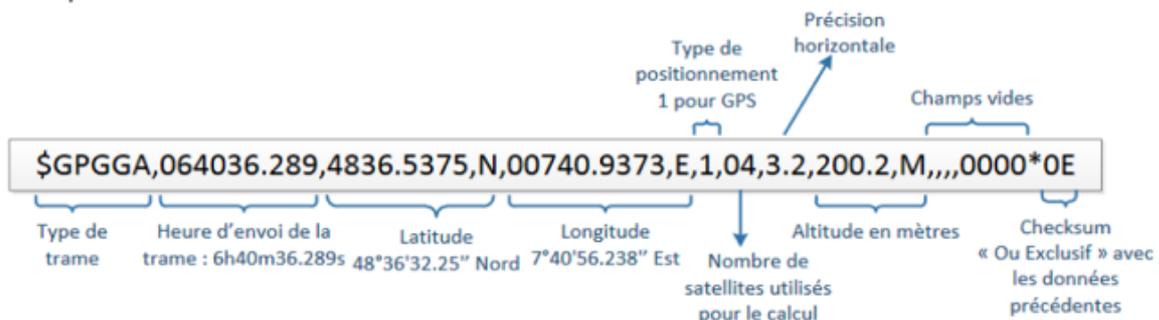
Sachant que l'onde se déplace à la vitesse de la lumière (environ 300 000 km/s) et que le message a été envoyé à 12h 36min 28s 12 et que le récepteur a reçu le message à 12h 36min 28s 19, quelle est la distance entre le récepteur et le satellite ?

Activité 2

Pour permettre à des périphériques informatiques de « dialoguer » entre eux, en réseau, il faut que les messages échangés respectent une syntaxe commune. On définit ce qu'on appelle un protocole, c'est à dire un ensemble de normes permettant à différents périphériques informatiques de dialoguer entre eux en réseau.

La norme NMEA 0183 est une spécification pour la communication entre équipements marins, dont les équipements GPS.

Exemple de trame GGA :



Exercice 1 :

On considère la trame suivante :

```
$GPGGA,153719.145, 4837.8332,N, 0448.8304,W,1,08,1.7,3.6,M, , , , *
```

- 1) Quelle est l'heure le récepteur GPS a-t-il enregistré cette position ?
- 2) A quelle altitude se situait le récepteur ?
- 3) Trouver les coordonnées géographiques du récepteur ?
- 4) A l'aide de OpenStretMap, identifier dans quelle ville se situait le récepteur au moment de cet enregistrement.

Activité 3

Récupérer les coordonnées GPS de la prise de vue photo_paysage.jpg à l'aide des méta données EXIF et du logiciel Gimp.

Puis identifier l'endroit où se situait le photographe au moment de prendre la photo.



c) Madame Y habite à Monbazillac en Dordogne. Elle possède un drone de loisir et souhaite le faire voler à une altitude d'environ 40 mètres. Indiquer si elle a le droit de le faire. Vous pouvez afficher la couche « restriction pour vol de drone » dans la rubrique Loisirs.

d) Déterminer l'itinéraire entre Le Havre et Rouen et donner la distance et la durée dans les cas suivant :

- A/ Itinéraire le plus rapide sans péage.
- B/ Itinéraire le plus court.
- C/ Itinéraire le plus rapide avec péage.

Itinéraire	A	B	C
Distance			
Durée			

Activité 5

Nous allons utiliser les cartes proposées par OpenStreetMap et le langage Python afin de générer des cartes personnalisées. Plus exactement, nous allons utiliser une bibliothèque Python nommée Folium qui va donc nous permettre de créer nos propres cartes à partir des cartes proposées par OpenStreetMap.

- Créer un dossier nommé CarteOSM
- Ouvrir Thonny et créer un nouveau programme nommé carte1.py
- Installer la bibliothèque 'folium'
- Recopier et exécuter le code suivant :

```
import folium

c=folium.Map(location=[46.078025, 6.409053])
c.save('maCarte.html')
```

- Aller dans le dossier CarteOSM, vous devriez voir la fichier "maCarte.html". Double-cliquez sur le fichier, votre navigateur web devrait afficher une carte centrée sur la ville de Bonneville. Notez bien que nous avons une véritable carte et pas une simple image (il est possible de zoomer ou de se déplacer).

Exercice 1 :

Ecrire un programme nommé 'carte2.py' qui génère une carte centrée sur le lycée Gambetta d'Arras..

Exercice 2 :

Reprendre la programme carte1.py et compléter le comme suit :

```
import folium

c=folium.Map(location=[46.078025, 6.409053], zoom_start=15)
c.save('maCarte.html')
```

Afin de vraiment personnaliser la carte, il est possible d'ajouter des marqueurs sur la carte. Un marqueur sera simplement défini par ses coordonnées GPS.

- Compléter le programme 'carte1.py'

```
import folium

c=folium.Map(location=[46.078025, 6.409053], zoom_start=20)
folium.Marker([46.078637266899, 6.4111924884134]).add_to(c)
c.save('maCarte.html')
```

Il est possible d'associer une information à un marqueur en ajoutant le paramètre "popup" Compléter le fichier :

```
import folium

c=folium.Map(location=[46.078025, 6.409053], zoom_start=20)
folium.Marker([46.078637266899, 6.4111924884134],popup="Lycée G Fichet").add_to(c)
c.save('maCarte.html')
```

Exercice 3 :

Compléter votre programme 'carte2.py' afin qu'il génère une carte sur le lycée Gambetta d'Arras (avec un zoom). Ajouter un 'Marker' et un 'popup' sur le lycée.

Activité 6

Le but de cette activité est de créer un programme Python qui permet d'afficher en temps réel, la position de la station spatiale internationale sur une carte OpenStreetMap.

Pour cela :

- Créer un dossier ISS
- Ouvrir Thonny et créer un nouveau programme nommé pos_iss.py
- Installer la bibliothèque 'requests'. Grâce à cette bibliothèque, nous allons récupérer le fichier json qui contient les coordonnées GPS de l'ISS.
- Recopier les lignes suivantes :

```
import requests
import folium

url = 'http://api.open-notify.org/iss-now.json'
content = requests.get(url)
data = content.json()

latitude = json_file['iss_position']['latitude']
longitude = json_file['iss_position']['longitude']
```

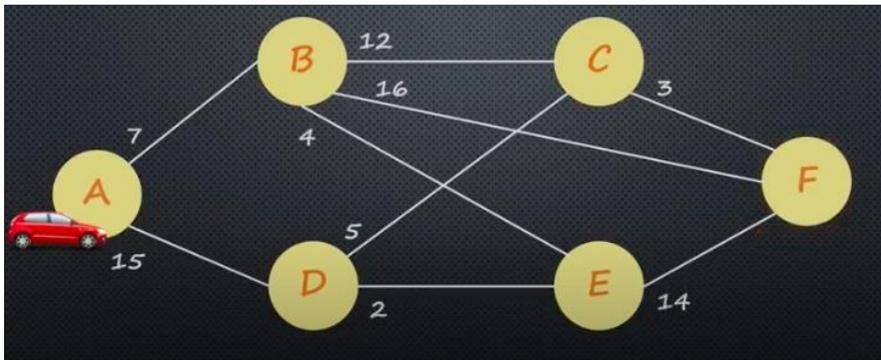
Vous venez de créer un programme qui récupère les coordonnées de l'ISS et qui les stocke dans les variables latitude et longitude.

En vous aidant des programmes précédents, compléter le programme afin qu'il génère une carte centrée sur l'ISS avec un 'Marker' et un 'popup'.

Activité 7

Déterminer le plus court chemin avec l'algorithme de DIJKSTRA. Voir la vidéo suivant pour comprendre l'algorithme : <https://www.youtube.com/watch?v=rHylCtXtdNs>

Appliquer cet algorithme avec le graphe suivant :



Déterminer le plus court chemin pour aller du sommet A au sommet F.