


Nom :

<b>ACADÉMIE de PARIS</b>	<i>Sciences Numériques et Technologie</i>	
		Seconde
	<b>TD / TP</b> <i>Une balade dans Paris, où comment l'IA Générative permet de produire un script Python pour répondre à un problème.</i>	2 Heures
<i>Lycée CHARLEMAGNE - 14 rue Charlemagne - 75004 PARIS</i>		

## Positionnement du problème à partir de l'existant

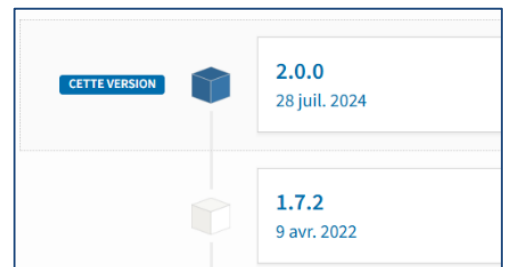
En 2020, dans le cadre des activités sur la géolocalisation en classe de seconde SNT, j'ai écrit un TP d'application en Python sur un calcul d'itinéraire qui se nomme « Une balade dans Paris »

Il avait pour but de faire afficher une carte permettant de tracer, à l'aide d'un navigateur WEB, l'itinéraire pour aller d'un point A à un point B au centre de Paris.

Les deux cas principaux d'étude étaient de déterminer d'une part l'itinéraire entre le parvis de la Cathédrale Notre Dame et l'entrée de la Grande Mosquée de Paris et d'autre part l'itinéraire entre le parvis de la Cathédrale Notre Dame et le parvis du Sacré Cœur dans le XVIII<sup>e</sup> arrondissement.

Les ressources utilisées étaient les bibliothèques Python Folium et Pyroutelib3 ainsi que le site Web OpenStreetMap.

En Juillet 2024, la bibliothèque Pyroutelib3 a subi une évolution majeure, rendant impossible l'exécution du code proposé sans modification importante de ce dernier.



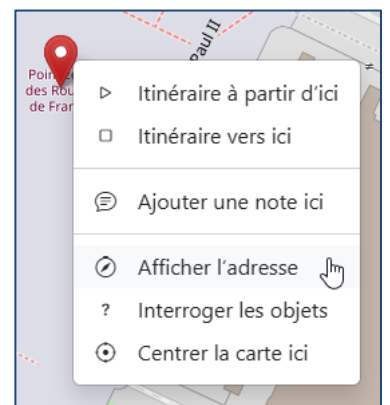
*L'objectif de ce nouveau TP est de repenser l'ancien en intégrant l'utilisation raisonnée d'une IA générative dans une approche Socratique.*

## I) Préliminaire : Récupération des points de géolocalisation sur le site OSM

Le site Open Street Map permet de récupérer des points géolocalisés à partir de leur nom ou de leur adresse. Il faut pour cela, par exemple, entrer une adresse puis sélectionner Afficher l'adresse à partir du menu contextuel.

✚ A l'aide d'un navigateur Web, aller sur le site <https://www.openstreetmap.org/>.

1. Sur le site OSM, rechercher la position GPS exacte du point O des routes de France (sur le Parvis de la Cathédrale Notre Dame de Paris.)
2. Sur le site OSM, rechercher la position GPS exacte de l'entrée de la Grande Mosquée de Paris dans le V arrondissement de Paris.
3. Sur le site OSM, rechercher la position GPS exacte du parvis du Sacré Cœur dans le XVIII<sup>e</sup> arrondissement de Paris.



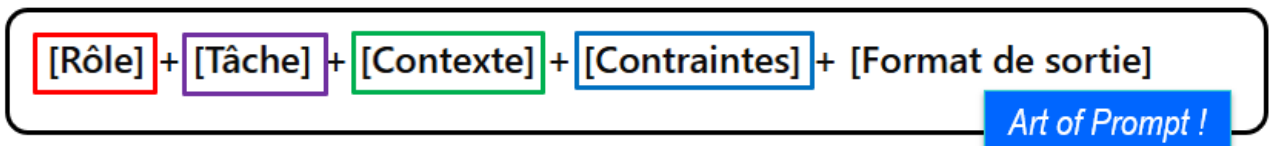
Nom :

## II) Découverte des librairies existantes par l'art du prompt

Cette partie du sujet a pour but de vous familiariser avec l'utilisation de l'IA dans un objectif d'apprentissage. Elle consiste à mettre en place des prompts dans une approche Socratique, c'est-à-dire dans une approche qui favorise grandement les interactions dans le but de comprendre et pas dans l'objectif d'obtenir une réponse toute faite. Elle est, elle-même basée, sur une notion plus large qui se nomme aujourd'hui *l'art de prompter* ou plus généralement *Art of Prompt*.

### Art of Prompt

L'université de Bordeaux propose un cours sur l'Art du Prompt et met en avant cinq parties différentes à faire apparaître dans un prompt efficace.



Note : Le point sur le format de sortie ne sera pas utile dans cette partie du sujet car notre utilisation de l'IA va se concentrer uniquement sur une conversation sans attendre un format spécifique dans la production d'un document final.

### Un premier prompt imposé

Pour répondre à la problématique de recherche de librairies satisfaisant la demande, le premier prompt est imposé. Il est donné dans l'encadré ci-dessous :

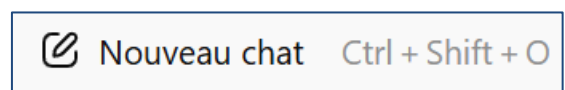
*Agis pour un élève de SNT en France en utilisant une approche Socratique et indique quelles sont les librairies Python utilisées classiquement pour rechercher un itinéraire à partir de cartes OSM.*

4. Dans le prompt précédent, encadrer en couleur les différentes parties du prompt.
  - ❖ En rouge, le rôle,
  - ❖ En violet, la tâche,
  - ❖ En vert, le contexte,
  - ❖ En bleu, la ou les contraintes.

✚ A l'aide d'un navigateur Web, aller sur le site <https://chatgpt.com/>

✚ Choisir Nouveau Chat.

✚ Dans la fenêtre de Dialogue, taper le prompt précédent.



### Exploitation d'un des résultats possibles

La caractéristique principale d'une conversation avec une IA générative est que le résultat obtenu n'est pas déterministe, il est donc normal qu'il varie entre dans le temps et dans l'espace (cela signifie que deux questions strictement identiques n'auront pas nécessairement la même réponse). Il faut retenir que cette particularité loin d'être une faiblesse est en réalité une force de ce type d'interaction entre l'homme et l'IA générative.

En Janvier 2025, une première partie de la réponse, donnée par ChatGPT est :

D'accord, je vais raisonner "comme un élève de SNT" qui cherche à comprendre en posant des questions, façon socratique, tout en donnant les vraies librairies Python classiques pour calculer des itinéraires à partir de données OpenStreetMap.

5. L'approche Socratique est-elle connue de l'IA ? Est-ce une approche par interactions de type Questions - Réponses ou une approche linéaire donnant simplement la réponse attendue ?

Nom :

✚ En Janvier 2025, la suite de cette réponse est :

### 🧠 Approche socratique (façon élève de SNT)

« Bon... si je veux trouver un itinéraire à partir d'une carte OSM, il me faut forcément deux choses, non ?

1. Les données de la carte (routes, intersections, etc.)
2. Un outil pour calculer un chemin dans ce réseau de routes.

Mais dans Python, quelles bibliothèques pourraient bien m'aider pour ça ?

Est-ce qu'il existe une bibliothèque qui peut *télécharger* les données OSM et aussi *tracer* des graphes ?

Et peut-être une autre qui peut *faire le calcul de plus court chemin* comme Dijkstra ?

Et est-ce que certaines réunissent tout ça d'un coup ? »

6. A l'issue de cette première partie de dialogue, y-a-t-il des questions plus précises qui apparaissent et finalement l'IA propose-t-elle une démarche dans la résolution du problème ?
7. De manière plus concrète, quelles sont les étapes à réaliser pour afficher l'itinéraire demandé ?

✚ En Janvier 2025, la suite de la réponse est :

1. OSMnx  
👉 La plus complète pour un usage pédagogique ou prototypage
2. NetworkX  
👉 Bibliothèque pour manipuler des graphes
3. Pyrosm  
👉 Import rapide de données OSM
4. osmnx + networkx "combo classique"
5. (optionnel, plus avancé) GraphHopper / OSRM via API

### 🌟 Résumé simple pour un élève de SNT

Si tu dois retenir **une seule bibliothèque**, c'est :

- ➡ **OSMnx**, parce qu'elle fait tout : récupérer les routes, créer le graphe, calculer l'itinéraire, afficher le résultat.

8. Quelles sont donc les deux bibliothèques à retenir ?

La conversation se termine sur une proposition.

Si tu veux, je peux aussi :

- t'aider à écrire un mini-exemple de calcul d'itinéraire avec OSMnx (sans dépasser le niveau SNT),
- t'expliquer comment fonctionne l'algorithme utilisé (Dijkstra, A\*...),
- ou te poser d'autres questions pour continuer le raisonnement socratique.

Tu veux aller vers quelle direction ?

Nom :

### III) Un second prompt imposé également et codage de la solution proposée

#### Un second prompt imposé

Bien que la dernière proposition indique clairement des directions possibles, le parti pris est d'imposer également le second prompt.

✚ Dans la fenêtre de Dialogue, taper le prompt suivant

*Toujours dans une approche Socratique et en agissant toujours pour un élève de seconde SNT en France, aide-moi à écrire un mini exemple de calcul d'itinéraire en utilisant le couple osmnx + networkx. Je souhaite un itinéraire partant du parvis de notre Dame de Paris pour se rendre à la grande Mosquée à pied.*

✚ En Janvier 2025, la première partie de la réponse est :

D'accord ! Je reste dans l'approche **socratique** (je te guide par questions comme si tu étais un élève de SNT qui cherche à comprendre par lui-même), et je t'aide à produire un **mini-exemple Python** avec **osmnx + networkx** pour calculer un itinéraire à pied entre :

- Le parvis de Notre-Dame de Paris
- La Grande Mosquée de Paris

#### 🧠 Approche socratique (façon élève SNT)

« Si je veux calculer un itinéraire avec Python, je dois d'abord me demander :

1. Comment récupérer la carte ?
2. Comment transformer ces routes en graphe ?
3. Comment trouver un chemin le plus court ?
4. Comment afficher le résultat ?

A titre d'exemple, deux exemples de code proposés par l'IA sont reproduits ci-dessous :

✚ Premier script

```
import osmnx as ox
import networkx as nx

print ('Debut 1')

# 1. On fixe les deux adresses
depart = "Parvis Notre-Dame, Paris, France"
arrivee = "Grande Mosquée de Paris, Paris, France"

# 2. On télécharge un graphe piéton autour de ces lieux
# 'walk' = réseau cyclable/piéton adapté à la marche
G = ox.graph_from_place("Paris, France", network_type="walk")

# 3. On convertit les adresses en coordonnées géographiques
coord_depart = ox.geocoder.geocode(depart)
coord_arrivee = ox.geocoder.geocode(arrivee)

# 4. On trouve les noeuds du graphe les plus proches de ces coordonnées
noeud_depart = ox.distance.nearest_nodes(G, coord_depart[1], coord_depart[0])
noeud_arrivee = ox.distance.nearest_nodes(G, coord_arrivee[1],
coord_arrivee[0])
```

Nom :

```
# 5. Calcul de l'itinéraire le plus court (en distance)
itineraire = nx.shortest_path(G, noeud_depart, noeud_arrivee, weight="length")

# 6. Affichage sur une carte
ox.plot_graph_route(G, itineraire, route_linewidth=3, node_size=0)

print('Fin 1')
```

#### 🚧 Second script

```
import osmnx as ox
import networkx as nx

print ('Debut 2')

# 1. Coordonées GPS des deux lieux
notre_dame = (48.8530, 2.3499)
mosquee = (48.8422, 2.3553)

# 2. Télécharger le graphe piéton autour de Paris
# (simplifié pour aller vite : distance = 2000 m autour de Notre-Dame)
G = ox.graph_from_point(notre_dame, dist=2000, network_type="walk")

# 3. Trouver les nœuds OSM les plus proches des deux points
start_node = ox.distance.nearest_nodes(G, notre_dame[1], notre_dame[0])
end_node = ox.distance.nearest_nodes(G, mosquee[1], mosquee[0])

# 4. Calculer le plus court chemin en distance
route = nx.shortest_path(G, start_node, end_node, weight='length')

# 5. Afficher le résultat
fig, ax = ox.plot_graph_route(G, route, route_linewidth=3, node_size=0)

print('Fin 2')
```

Note importante : Dans les deux scripts, a été ajouté des commandes d'affichage indiquant les Débuts et Fins de chaque sous-ensemble (surlignés dans les deux exemples de scripts précédents) afin d'apporter de la clarté lors de l'exécution des programmes.

9. Par l'analyse du code proposé, répondre alors aux questions qui sont posées dans le cadre conversationnel avec l'IA et qui sont reproduites ci-dessous :
  - ❖ Comment récupérer la carte ?
  - ❖ Comment transformer ces routes en graphe ?
  - ❖ Comment trouver un chemin le plus court ?
  - ❖ Comment afficher le résultat ?
10. Quelle est l'information présente dans le code indiquant que la recherche de l'itinéraire est faite pour un trajet pédestre ?

### Codage de la solution proposée

- Ouvrir un Environnement de Développement Intégré comme Spyder (The Scientific Python Development Environment) ou Jupyter ou Capytale.
- Créer un nouveau projet et recopier le code proposé par l'IA.
- Lancer le script par la commande Run.

Attention : Le script est nécessairement long à s'exécuter et il convient d'être patient.

Nom :

11. Indiquer quelle est la raison principale de cette durée d'exécution ?
12. Vérifier que la carte obtenue correspond bien à votre demande.

**Faire valider votre travail par l'enseignant :**

Oui  : Non  : Aide

### Des prompts libres mais des contraintes imposées

A partir des exemples précédents, on désire désormais obtenir deux itinéraires différents pour compléter nos balades dans Paris.

- + Un premier itinéraire reliant le parvis de Notre Dame à celui du Sacré Cœur.
- + Un second reliant le Parvis de Notre Dame à l'entrée du Jardin du Luxembourg sur la rue de Vaugirard.

Pour les deux premiers cas, les contraintes respectives sont :

- + Pour la première :
  - o Fond de carte blanc ; Chemin en rouge ; Ecrire « Itinéraire vers le Sacré Cœur » sur la carte (en haut à droite)
- + Pour la seconde :
  - o Fond de carte blanc ; Chemin en bleu ; Ecrire « Itinéraire vers le Luxembourg » sur la carte (en bas à gauche)

13. Pour les deux premiers cas, proposer deux prompts distincts permettant d'obtenir les propositions de script.

- ✓ Dans l'Environnement de Développement Intégré, recopier chaque script proposé en prenant soin au préalable de faire afficher Début et Fin comme dans la version proposée précédemment.

Note importante : Il est fortement conseillé de placer les scripts précédents en commentaire car le temps d'exécution est long et il convient de se focaliser uniquement sur cette partie du sujet sans reprendre la partie précédente.

14. Après avoir vérifié que l'exécution de chaque script correspond bien aux consignes imposées, recopier dans votre compte rendu les scripts proposées et les cartes obtenues.

**Faire valider votre travail par l'enseignant :**

Oui  : Non  : Aide

### Pour aller plus loin – 1 –

En autonomie, faire afficher la distance de ces deux trajets.

Si vous ne savez pas faire, ce qui est bien normal, il faut alors le demander à l'IA en Approche Socratique et bien entendu comprendre et mettre en œuvre la réponse proposée !

**Faire valider votre travail par l'enseignant :**

Oui  : Non  : Aide

### Pour aller plus loin – 2 –

En autonomie, proposer un troisième itinéraire laissé à votre discrétion. Pour ce troisième cas, les contraintes supplémentaires sont laissées libres.

**Faire valider votre travail par l'enseignant :**

Oui  : Non  : Aide