

DS d'informatique n° 5  
le Samedi 8 Juin 2024 - durée 1h.

On dispose des lignes de métro de Paris sous la forme des listes des gares desservies dans l'ordre. (On ne tient pas compte des lignes qui se divise en deux)

```
ligne1 = ['La defense', ..., 'Louvre-Rivoli', 'Chatelet', 'Hotel de Ville',\n         ..., 'Chateau de Vincennes']\nligne2= ['Porte Dauphine', ..., 'Nation']\n...\nligne14 = ['Mairie de Saint-Ouen', ..., 'Olympiades']\nmetro = [ligne1, ligne2, ..., ligne14]
```

La liste `metro` regroupe ainsi tout le réseau du métro parisien.

On recherche à construire un algorithme pour rechercher son chemin dans le métro.

1. On commence par construire un graphe  $G$  dont les sommets sont les gares. Ecrire une fonction `init_sommet(metro)` qui renvoie  $S$  la listes des gares (sans doublons) et triées par ordre alphabétique.

Les fonctions **set** et **sort** sont ici interdites ou devront être reprogrammer.

2. Les arrêtes pondérées sont les trajets réalisés entre deux gares consécutives d'une même ligne. L'étiquette est le numéro de la ligne. Par exemple, il existe une arrête entre 'Louvre-Rivoli' et 'Chatelet' d'étiquette 1. Il n'y pas d'arrête entre 'Mairie de Saint-Ouen' et 'Olympiades' donc on pourra associée la valeur **None**.

Chacune des gares est désormais associées à son indice dans la liste  $S$  créé. La fonction `index` est ici interdite ou devra être reprogrammée.

Ecrire une fonction `init_arrete(metro)` qui renvoie la matrice d'adjacence associée aux arrêtes de  $G$ .

3. Ecrire un programme de parcours en largeur qui recherche un chemin entre deux gares `chemin(depart, arrive)`. On renverra la liste `lgare` des indices des gares.

On pourra utiliser `file=collections.deque()` pour créer une file avec Python.

4. Ecrire un programme `trajet(lgare)` qui renvoie `(lnum,lnom)` la liste des numéros des lignes à emprunter et le nom des gares où il faut changer de train.
5. Expliquer pourquoi on a ici préférer un parcours en largeur plutôt qu'en profondeur. Quelle est la 'quantité' qui est ainsi minimale ?
6. Quelle serait l'avantage par rapport à cette méthode d'utiliser l'algorithme de Dijkstra ? Pourquoi les données proposées par l'énoncé ne seraient pas suffisante pour utiliser l'algorithme A\* ?