MINI PROJET :
METRO PARIS – NYC

Description :

Travail sur les structures alternatives, itératives, POO et la bibliothèque *Pandas*.

Ce mini-projet peut être traité seul ou par un groupe de deux personnes.

Il sera rendu sur MBN.
Le nom du fichier sera du type "*Nom1-Nom2.py*" et les deux noms des élèves devront figurer dans les commentaires de début de programme.

Le but du projet est de pouvoir trouver le plus court chemin d'une station de métro à une autre à l'aide de l'objet **Graphe** du fichier *dijkstra.py* où est implémenté l'algorithme de Dijkstra que l'on utilisera en *module*.

On le testera sur le métro de Paris (RATP) et le métro de New York City (MTA).

RATP
(Régie Autonome des Transports Parisiens)

MTA
(Metropolitan Transportation Authority)

Les plans des métros sont donnés en ressources au format *.pdf* et sont nommés *Plan_Metro_Ville.pdf*.

Pour chaque métro, on possède deux fichiers d'attributs identiques :

- Le fichier *…_nodes.csv* décrit les stations.
- Le fichier *…_edges.csv* décrit les liaisons entre les stations.

ratp_nodes.csv

```
id;name;line;lat;long
0;Abbesses;12;48.8843997149084;2.33839938417104
1;Alexandre Dumas;2;48.8564263483252;2.39455425765499
2;Alma Marceau;9;48.8642986710467;2.30125141044993
3;Alésia;4;48.8280660196865;2.32682742005084
4;Anatole France;3;48.8920186263289;2.28551744479448
5;Anvers;2;48.8828716901806;2.34416357280039
6;Argentine;1;48.8756724859879;2.28944416448145
7;Arts et Métiers;11;48.8653811760469;2.35564493120688
8;Arts et Métiers;3;48.8653811760469;2.35564493120688
```

mta_nodes.csv

```
id;name;line;lat;long
101;Van Cortlandt Park - 242 St;1;40.889248;-73.898583
103;238 St;1;40.884667;-73.90087
103;238 St;1;40.884667;-73.90087
104;231 St;1;40.878856;-73.904834
104;231 St;1;40.878856;-73.904834
106;Marble Hill - 225 St;1;40.874561;-73.909831
106;Marble Hill - 225 St;1;40.874561;-73.909831
107;215 St;1;40.869444;-73.915279
107;215 St;1;40.869444;-73.915279
```

Les attributs de *…_nodes.csv* sont :

- id : numéro identifiant le station (clé primaire).
- name : nom de la station.
- line : ligne sur laquelle se trouve la station.
- lat : latitude de la station en degré.
- long : longitude de la station en degré.

Attention : Une station peut se trouver sur plusieurs lignes (connexion) et possède donc le même nom mais un id différent (exemple de la station *Arts et Métiers* dans le fichier *ratp_nodes.csv*).

ratp_edges.csv

```
id1;id2;time;connexion
0;159;46;Route
0;238;41;Route
1;12;36;Route
1;235;44;Route
2;110;69;Route
2;139;50;Route
3;210;41;Route
3;262;33;Route
4;171;43;Route
```

mta_edges.csv

```
id1;id2;time;connexion
101;103;90;Route
103;101;90;Route
103;104;90;Route
104;103;90;Route
104;106;90;Route
106;104;90;Route
106;107;90;Route
107;106;90;Route
107;108;60;Route
```

Les attributs de *…_edges.csv* sont :

- id1 : numéro identifiant la première station.
- id2 : numéro identifiant la deuxième station.
- time : temps en secondes entre les deux stations.
- connexion : station sur une ligne (Route) ou sur plusieurs lignes (Transfer)

La bibliothèque Pandas :

La bibliothèque *pandas* permet de créer des tableaux sous forme de DataFrame, structure de données correspondant à un *tableau 2D*.

- La méthode ***read_csv(fichier,separateur)*** permet de transformer un fichier *.csv* en DataFrame :

df = ***read_csv***(*ratp_nodes.csv,sep=' ; '*)

ratp_nodes.csv

```
id;name;line;lat;long
0;Abbesses;12;48.8843997149084;2.33839938417104
1;Alexandre Dumas;2;48.8564263483252;2.39455425765499
2;Alma Marceau;9;48.8642986710467;2.30125141044993
3;Alésia;4;48.8280660196865;2.32682742005084
4;Anatole France;3;48.8920186263289;2.28551744479448
5;Anvers;2;48.8828716901806;2.34416357280039
6;Argentine;1;48.8756724859879;2.28944416448145
7;Arts et Métiers;11;48.8653811760469;2.35564493120688
8;Arts et Métiers;3;48.8653811760469;2.35564493120688
```

df_sommets (DataFrame)

	id	name	line	lat	long
0	0	Abbesses	12	48.884400	2.338399
1	1	Alexandre Dumas	2	48.856426	2.394554
2	2	Alma Marceau	9	48.864299	2.301251
3	3	Alésia	4	48.828066	2.326827
4	4	Anatole France	3	48.892019	2.285517
5	5	Anvers	2	48.882872	2.344164
6	6	Argentine	1	48.875672	2.289444
7	7	Arts et Métiers	11	48.865381	2.355645
8	8	Arts et Métiers	3	48.865381	2.355645

La 1ère colonne correspond à la numérotation automatique des lignes.

- La méthode ***set_index( )*** permet de remplacer la numérotation automatique des lignes par une des colonnes du DataFrame :

df_noms = *df_sommets*.***set_index***('*name*')

df_sommets (DataFrame)

	id	name	line	lat	long
0	0	Abbesses	12	48.884400	2.338399
1	1	Alexandre Dumas	2	48.856426	2.394554
2	2	Alma Marceau	9	48.864299	2.301251
3	3	Alésia	4	48.828066	2.326827
4	4	Anatole France	3	48.892019	2.285517
5	5	Anvers	2	48.882872	2.344164
6	6	Argentine	1	48.875672	2.289444
7	7	Arts et Métiers	11	48.865381	2.355645
8	8	Arts et Métiers	3	48.865381	2.355645

df_noms (DataFrame)

name	id	line	lat	long
Abbesses	0	12	48.884400	2.338399
Alexandre Dumas	1	2	48.856426	2.394554
Alma Marceau	2	9	48.864299	2.301251
Alésia	3	4	48.828066	2.326827
Anatole France	4	3	48.892019	2.285517
Anvers	5	2	48.882872	2.344164
Argentine	6	1	48.875672	2.289444
Arts et Métiers	7	11	48.865381	2.355645
Arts et Métiers	8	3	48.865381	2.355645

- La navigation dans les DataFrames se fait comme dans une liste de liste, en indiquant le nom de la colonne, puis le numéro de la ligne :

df_sommets['name'][6] = '*Argentine*'

- La méthode *df.to_dict()* permet de transformer un DataFrame en dictionnaire dont les clés sont les noms des colonnes et la valeur un dictionnaire dont les clés sont la 1ère colonne et la valeur de la ligne :

```
{'id': {0: 0, 1: 1, 2: 2, 3: 3, 4: 4, 5: 5, 6: 6, 7: 7, 8: 8},
 'lat': {0: 48.8843997149084,
         1: 48.8564263483252,
         2: 48.8642986710467,
         3: 48.8280660196865,
         4: 48.8920186263289,
         5: 48.8828716901806,
         6: 48.8756724859879,
         7: 48.8653811760469,
         8: 48.8653811760469},
 'line': {0: 12, 1: 2, 2: 9, 3: 4, 4: 3, 5: 2, 6: 1, 7: 11, 8: 3},
 'long': {0: 2.33839938417104,
          1: 2.39455425765499,
          2: 2.30125141044993,
          3: 2.32682742005084,
          4: 2.28551744479448,
          5: 2.34416357280039,
          6: 2.28944416448145,
          7: 2.35564493120688,
          8: 2.35564493120688},
 'name': {0: 'Abbesses',
          1: 'Alexandre Dumas',
          2: 'Alma Marceau',
          3: 'Alésia',
          4: 'Anatole France',
          5: 'Anvers',
          6: 'Argentine',
          7: 'Arts et Métiers',
          8: 'Arts et Métiers'}}
```

df_sommets.to_dict()

```
{'id': {'Abbesses': 0,
        'Alexandre Dumas': 1,
        'Alma Marceau': 2,
        'Alésia': 3,
        'Anatole France': 4,
        'Anvers': 5,
        'Argentine': 6,
        'Arts et Métiers': 8},
 'lat': {'Abbesses': 48.8843997149084,
         'Alexandre Dumas': 48.8564263483252,
         'Alma Marceau': 48.8642986710467,
         'Alésia': 48.8280660196865,
         'Anatole France': 48.8920186263289,
         'Anvers': 48.8828716901806,
         'Argentine': 48.8756724859879,
         'Arts et Métiers': 48.8653811760469},
 'line': {'Abbesses': 12,
          'Alexandre Dumas': 2,
          'Alma Marceau': 9,
          'Alésia': 4,
          'Anatole France': 3,
          'Anvers': 2,
          'Argentine': 1,
          'Arts et Métiers': 3},
 'long': {'Abbesses': 2.33839938417104,
          'Alexandre Dumas': 2.39455425765499,
          'Alma Marceau': 2.30125141044993,
          'Alésia': 2.32682742005084,
          'Anatole France': 2.28551744479448,
          'Anvers': 2.34416357280039,
          'Argentine': 2.28944416448145,
          'Arts et Métiers': 2.35564493120688}}
```

df_sommets.set_index('name').to_dict()

Production et présentation :

Votre travail consiste, en utilisant l'implémentation de l'algorithme de Dijkstra, à déterminer le plus court chemin entre deux stations du métro de Paris et de New York City.

- Créer une classe **Metro** ayant les attributs passés en paramètres du constructeur :
  - *fichier_sommets* : nom du fichier contenant les informations sur les sommets (str)
  - *fichier_arcs* : nom du fichier contenant les informations sur les arêtes (str)

- Ajouter à la classe **Metro** les attributs :
  - *sommet_ligne* : dictionnaire donnant la *ligne* d'une station connaissant son *id*, initialiser à vide (dic)
  - *sommet_id* : dictionnaire donnant le *nom* d'une station connaissant son *id*, initialiser à vide (dic)
  - *sommet_nom* : dictionnaire donnant l'*id* d'une station connaissant son *nom*, initialiser à vide (dic)

- Créer une méthode ***fabrication_graphe( )*** qui fabrique le graphe à partir de l'objet **Graphe** du module *dijkstra* et de l'attribut *fichier_arcs* :
  - *None*
  - **Retour** : None

  On pourra itérer dans un DataFrame *df_arcs* créé à partir de l'attribut *fichier_arcs* et de la méthode ***read_csv( )*** de la bibliothèque *Pandas*.

df_arcs (DataFrame)

	id1	id2	time	connexion
0	0	159	46	Route
1	0	238	41	Route
2	1	12	36	Route
3	1	235	44	Route
4	2	110	69	Route
5	2	139	50	Route
6	3	210	41	Route
7	3	262	33	Route
8	4	171	43	Route

- Créer une méthode ***recuperation_donnees( )*** qui fabrique les attributs suivants à partir de l'attribut *fichier_sommets* :
  - *sommet_ligne* : dictionnaire donnant la *ligne* d'une station connaissant son *id*.

```
{0: 12, 1: 2, 2: 9, 3: 4, 4: 3, 5: 2, 6: 1, 7: 11, 8: 3}
```

- *sommet_id* : dictionnaire donnant le *nom* d'une station connaissant son *id*.

```
{0: 'Abbesses',
 1: 'Alexandre Dumas',
 2: 'Alma Marceau',
 3: 'Alésia',
 4: 'Anatole France',
 5: 'Anvers',
 6: 'Argentine',
 7: 'Arts et Métiers',
 8: 'Arts et Métiers'}
```

- *sommet_nom* : dictionnaire donnant l'*id* d'une station connaissant son *nom*.

```
{'Abbesses': 0,
 'Alexandre Dumas': 1,
 'Alma Marceau': 2,
 'Alésia': 3,
 'Anatole France': 4,
 'Anvers': 5,
 'Argentine': 6,
 'Arts et Métiers': 8}
```

- Créer une méthode ***itineraire_dijkstra(****depart,arrivee****)*** qui renvoie l'itinéraire (avec les lignes de métro) et le temps :
  - *depart* : station de départ (str)
  - *arrivee* : station d'arrivée (str)

  - **Retour** : le temps pour aller du départ à l'arrivée et le chemin correspond (liste de tuple comprenant les stations et les lignes)

Exemple : Chemin pour aller de *Pigalle* à *Place d'Italie* à Paris

Pigalle

Place d'Italie


```
(1424,
 [('Pigalle', 2),
  ('Anvers', 2),
  ('Barbès Rochechouart', 2),
  ('Barbès Rochechouart', 4),
  ('Gare du Nord', 4),
  ("Gare de l'Est", 4),
  ("Château d'Eau", 4),
  ('Strasbourg-Saint-Denis', 4),
  ('Réaumur-Sébastopol', 4),
  ('Étienne Marcel', 4),
  ('Les Halles', 4),
  ('Châtelet', 4),
  ('Châtelet', 7),
  ('Pont-Marie', 7),
  ('Sully Morland', 7),
  ('Jussieu', 7),
  ('Place Monge', 7),
  ('Censier Daubenton', 7),
  ('Les Gobelins', 7),
  ("Place d'Italie", 7)])
```

Exemple : Chemin pour aller du *81 St - Museum of Natural History* à *South Ferry* à NYC

81 St - Museum of Natural History

South Ferry

```
(1440,
 [('81 St - Museum of Natural History', 'C'),
  ('72 St', 'C'),
  ('59 St - Columbus Circle', 'D'),
  ('59 St - Columbus Circle', '2'),
  ('50 St', '2'),
  ('Times Sq - 42 St', '3'),
  ('34 St - Penn Station', '3'),
  ('14 St', '3'),
  ('Chambers St', '3'),
  ('Cortlandt St', '1'),
  ('Rector St', '1'),
  ('South Ferry', '1')])
```

Amélioration possible :

- Créer une méthode ***affichage( )*** qui permet l'affichage ci-dessous :

```
Durée : 00:23:44

Pigalle                 : ligne 2
Anvers                  : ligne 2
Barbès Rochechouart     : ligne 2
Barbès Rochechouart     : ligne 4
Gare du Nord            : ligne 4
Gare de l'Est           : ligne 4
Château d'Eau           : ligne 4
Strasbourg-Saint-Denis  : ligne 4
Réaumur-Sébastopol      : ligne 4
Étienne Marcel          : ligne 4
Les Halles              : ligne 4
Châtelet                : ligne 4
Châtelet                : ligne 7
Pont-Marie              : ligne 7
Sully Morland           : ligne 7
Jussieu                 : ligne 7
Place Monge             : ligne 7
Censier Daubenton       : ligne 7
Les Gobelins            : ligne 7
Place d'Italie          : ligne 7
```

- Trouver l'itinéraire du métro pour aller du premier site au deuxième à Paris :

- Trouver l'itinéraire du métro pour aller du premier site au deuxième à NYC :

- ...