



# Exercices sur les graphes

## S4 - Arbres et graphes

Les exercices précédés du symbole  sont à faire sur machine, en sauvegardant le fichier si nécessaire.

Les exercices précédés du symbole  doivent être résolus par écrit.

### Exercice 1

On dit qu'un graphe est **simple** s'il n'y a pas de boucle ni de double arête. Un graphe simple est dit **complet** s'il est non orienté et que chaque paire de sommets est reliée par une arête.

Dessiner un graphe complet d'ordre 3, puis un graphe complet d'ordre 4.

### Exercice 2

#### Graphe d'un réseau électrique

Un de vos amis travaille pour un distributeur d'électricité.

Il doit proposer à son supérieur une représentation du réseau reliant différentes villes. Comme il n'y arrive pas trop, il voudrait que vous la lui fassiez.

Pour simplifier le problème, il a déjà renommé les villes en A, B, C, D, E et F. De plus, il vous donne les informations suivantes :

- la ville B est reliée par un réseau électrique aux villes A, C et D,
- la ville A est reliée par un réseau électrique aux villes B, E et F,
- la ville C est reliée par un réseau électrique aux villes B, D, E et F,
- la ville D est reliée par un réseau électrique aux villes B, C et F,
- la ville E est reliée par un réseau électrique aux villes A, C et F,
- la ville F est reliée par un réseau électrique aux villes A, C, D et E.

Proposer un graphe qui modélise la situation.

Ce graphe est-il complet ? Pourquoi ?

Donner sa matrice d'adjacence.

### Exercice 3

Voici un ensemble de relations :

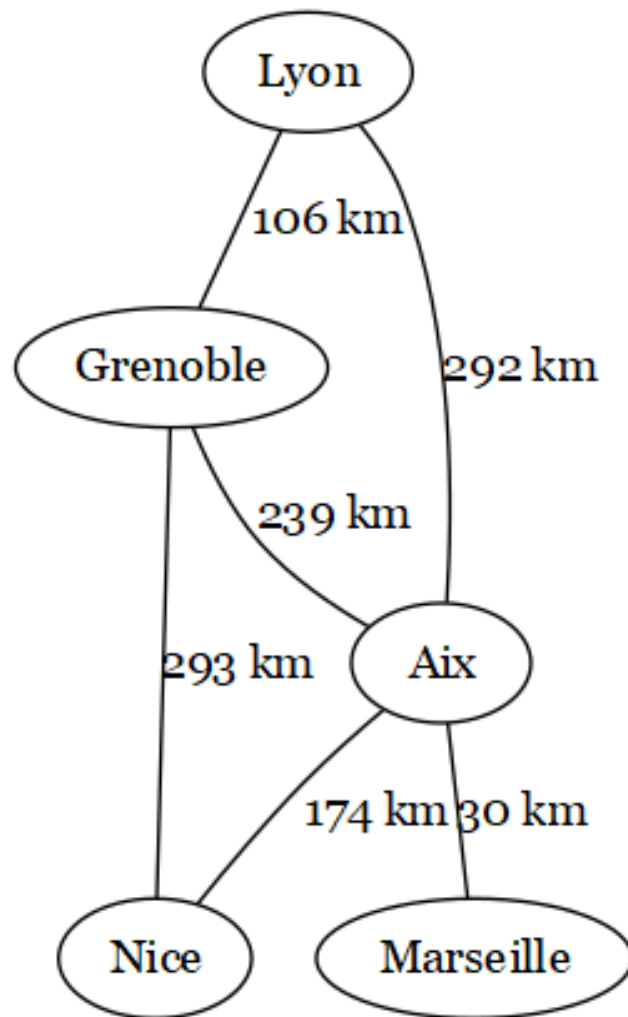
- A est ami avec tout le monde sauf G,
- B est ami avec A, D et H,
- C est ami avec A, F, G et H,
- D est ami avec A, B et H,
- E est ami avec A et H,
- F est ami avec A et C,
- G est ami avec C et H,
- H est ami avec A, B, C, D, E et G.

1. Représenter ce graphe et vérifier qu'il est non orienté.
2. Implémenter ce graphe sous la forme d'une liste d'adjacence dans laquelle chaque clé représente le nœud étudié.
3. Écrire la matrice d'adjacence et vérifier qu'elle est symétrique (on utilisera l'ordre alphabétique pour indexer les nœuds).
- 4 Implémenter la matrice en python sous la forme d'une liste de liste.

### Exercice 4

Voici un graphe représentant les distances dans le réseau sud-est.

1. Implémenter ce graphe sous la forme d'un dictionnaire de liste dans lequel chaque clé représente le nœud étudié et les sommets adjacents sont représentés sous la forme d'une liste de dictionnaires clé(sommet adjacent): valeur(distance).
2. Écrire la matrice d'adjacence et vérifier qu'elle est symétrique(on utilisera l'ordre alphabétique pour indexer les nœuds).
3. Proposer un algorithme qui renvoie tous les chemins possibles pour aller de Nice à Lyon sans jamais passer deux fois par la même ville. On utilisera trois paramètres d'entrée: le graphe implémenté sous la forme d'un dictionnaire d'adjacence comme celui de la question 2 et les deux villes de départ et d'arrivée.



Données issues de geoportail avec l'itinéraire le plus court

Figure 1: Réseau sud-Est