

# XIX – Les graphes

## 1/ Introduction

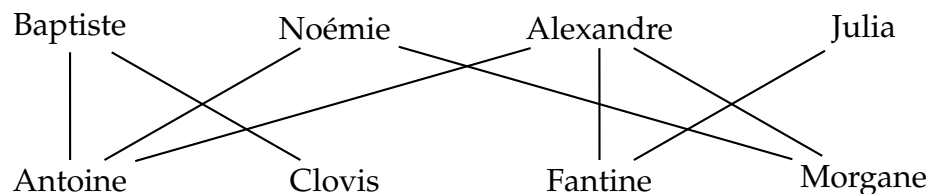
La théorie des graphes, qui remonte aux travaux d'Euler au XVIII<sup>e</sup> siècle, trouve son origine dans l'étude de problèmes combinatoires comme le problème des ponts de Königsberg, le déplacement d'un cavalier sur un échiquier ou le coloriage de cartes. Elle constitue une branche à part entière des mathématiques pures et de l'informatique théorique, et elle est utilisée dans diverses disciplines.

Les graphes interviennent en effet dans nombre d'algorithmes répondant à des problématiques concrètes, que ce soit en biologie, en sciences sociales ou dans l'industrie des télécommunications. Leur étude s'est considérablement développée depuis le XX<sup>e</sup> siècle et constitue une mine inépuisable de sujets de concours !

Un **graphe** est un ensemble de points reliés par des **arêtes**, munies éventuellement de **poids**. Il permet de représenter la structure et les connexions d'un ensemble complexe en exprimant les relations entre ses éléments (réseaux de télécommunications, interactions de diverses espèces animales, circuits électriques, etc...)

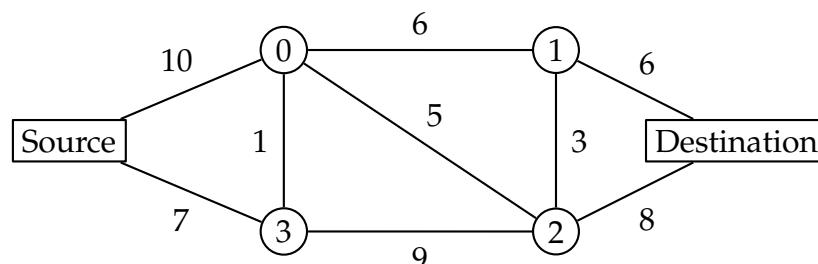
À titre d'exemple, mentionnons trois utilisations classiques des graphes.

**Réseaux d'amitié :** un graphe permet de visualiser les liens d'amitiés entre les individus d'un groupe donné. On peut ainsi étudier les groupes où tout le monde se connaît.



**Plus court chemin :** on peut représenter un réseau routier à l'aide d'un graphe et ainsi déterminer le plus court chemin d'un point à un autre.

**Réseau informatique :** le graphe représente les flux d'information qui passent à travers différents points du réseau. Il permet d'étudier le flux maximal d'information pouvant transiter entre une source et une destination.



## 2/ Définitions

### 2.1) Sommets, arcs et arêtes

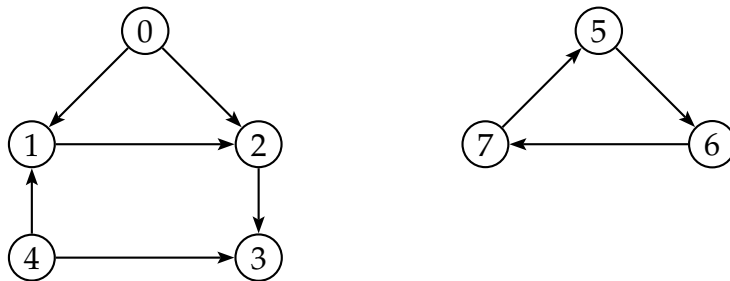
**Définition** Un **graphe (orienté)** est un couple  $(S, A)$ , où  $S$  est un ensemble et  $A$  une partie de  $S^2$ .

- Les éléments de  $S$  sont appelés les **sommets** ou **nœuds** du graphe.
- Les éléments de  $A$  sont appelés les **arcs** (ou **arêtes**) du graphe.

**Remarque :** pour le moment, on supposera que  $S = \llbracket 0 ; N - 1 \rrbracket$  où  $N = \text{card}(S)$ .

#### Représentation graphique

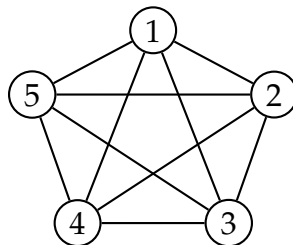
Sur un dessin, les sommets du graphe sont représentés par des cercles contenant leur numéro et les arcs par des flèches. Ainsi, l'arc  $(i, j)$  est représenté par une flèche reliant le sommet  $i$  au sommet  $j$ . Voici un très bel exemple de graphe orienté :



**Définition** Soit  $G = (S, A)$  un graphe.

- On dit que  $G$  est **complet** si  $(i, j) \in A$  pour tous sommets  $i \neq j \in S$ .
- On dit que  $G$  est **non orienté** si  $(j, i) \in A$  pour tout arc  $(i, j) \in A$ .  
Ses arcs sont alors appelés **arêtes** et on les représente à l'aide d'un simple trait.

**Remarque :** les arêtes d'un graphe non-orienté sont des parties à deux éléments de  $S$ . Un tel graphe se représente comme un graphe orienté, en remplaçant chaque arête  $\{i, j\}$  par un couple d'arcs  $(i, j)$  et  $(j, i)$ , et réciproquement. Voici un graphe non orienté complet :



**Définition** Soient  $G = (S, A)$  un graphe et  $i \in S$ .

- On appelle **degré sortant de  $i$**  le nombre  $d_+(i)$  de sommets  $j \in S$  tels que  $(i, j) \in A$ .
- On appelle **degré entrant de  $i$**  le nombre  $d_-(i)$  de sommets  $j \in S$  tels que  $(j, i) \in A$ .
- Si le graphe est non-orienté, les degrés entrant et sortant de  $i$  sont égaux : on parle simplement du **degré de  $i$**  que l'on note  $d(i)$ .

**Définition** Soit  $G = (S, A)$  un graphe.

- Deux sommets sont **adjacents** s'il existe une arête ou un arc les reliant.
- On appelle **boucle** une arête ou un arc reliant un sommet à lui-même.
- Si  $(i, j) \in A$ , on dit que  $j$  est le **successeur** ou le **voisin** de  $i$ .

**Remarque :** on parle aussi de sommets antécédents et descendants, pères et fils, etc...

Dans la suite, on considérera des graphes **simples**, dans lesquels on ne retrouve pas deux fois le même arc ou la même arête, et sans boucles.

## 2.2) Chemins et connexité

**Définition** Soient  $G = (S, A)$  un graphe,  $p \in \mathbb{N}^*$  et  $s_0, s_1, \dots, s_p$  des éléments de  $S$ .

- On dit que la suite  $(s_0, \dots, s_p)$  est un **chemin** du graphe  $G$  si l'on a  $(s_i, s_{i+1}) \in A$  pour tout  $i \in \llbracket 0; p-1 \rrbracket$ . L'entier  $p$  est alors la **longueur** du chemin.
- Ce chemin est appelé **circuit** lorsque  $s_0 = s_p$ .
- Si  $s_i \neq s_j$  pour tout  $i \neq j$ , on dit que le chemin est **élémentaire**.
- Dans le cas d'un circuit, on parle de **circuit élémentaire** lorsque  $(s_0, \dots, s_{p-1})$  est un chemin élémentaire.

**Remarque :** dans un graphe non orienté, les circuits sont appelés **cycles** et les chemins constitués d'arêtes distinctes sont appelés **chaînes**.

**Exemple :** dans le premier graphe, le chemin  $(5, 6, 7, 5)$  est un circuit élémentaire.

**Définition** Soient  $G = (S, A)$  un graphe. On définit la relation binaire  $\mathcal{R}$  sur  $S$  par  $u\mathcal{R}v$  si, et seulement si, il existe un chemin de  $u$  vers  $v$  et un chemin de  $v$  vers  $u$ .

**Remarque :** pour un graphe non-orienté, cela revient à dire qu'il existe un chemin reliant le sommet  $u$  au sommet  $v$ . C'est le cas pour tous les sommets du second graphe : comme il est complet, on a des chemins de longueur 1 entre tous les sommets.

S'il existe un chemin de  $u$  à  $v$  et un chemin de  $v$  à  $w$ , on obtient alors en les concaténant un chemin de  $u$  à  $w$  dont la longueur est la somme des deux premiers. On a même mieux :

**Proposition**

*La relation  $\mathcal{R}$  est une relation d'équivalence sur  $S$  dont les classes d'équivalences sont appelées les **composantes fortement connexes** du graphe.*

**Remarque :** pour un graphe non-orienté, on parle simplement de **composantes connexes**.

En d'autres termes, une composante (fortement) connexe d'un graphe est sous-ensemble de sommets maximal au sens de l'inclusion, pour lequel on peut toujours trouver un chemin reliant deux sommets quelconques.

**Exemple :** les composantes du premier graphe sont  $\{0\}, \{1\}, \{2\}, \{3\}, \{4\}, \{5, 6, 7\}$ .

**Proposition**

- Les composantes (fortement) connexes d'un graphe forment une **partition** de celui-ci.
- Si le graphe en possède une seule, il est dit (**fortement**) **connexe**.

**Exemple** : le second graphe est connexe ; c'est le cas de tout graphe complet.

**2.3) Graphes pondérés**

**Définition** Un graphe  $(S, A)$  est dit **pondéré** ou **valué** s'il est muni d'une fonction dite **de poids** définie sur  $A$  et à valeurs dans  $\mathbb{R}$ .

**Remarques :**

- ❶ La plupart du temps, on donne cette fonction sur le dessin en plaçant le poids de chaque arête à côté de celle-ci (cf. l'exemple vu en introduction).
- ❷ Les graphes pondérés vont permettre de représenter des réseaux routiers, les sommets correspondant aux villes et les poids aux distances.
- ❸ On procèdera de même pour modéliser un réseau informatique, les poids correspondant au flux maximal d'information pouvant passer par chaque branche du réseau.

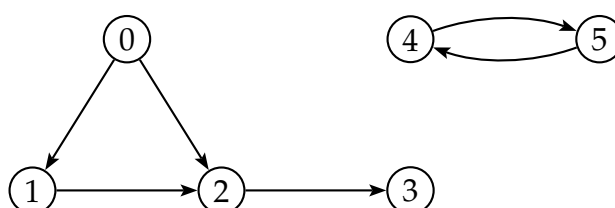
**3/ Représentation des graphes****3.1) Différents types de représentation**

En supposant toujours que les nœuds du graphe sont les entiers de 0 à  $N - 1$ , on a trois façons de le représenter :

- ❶ On considère la liste constituée de  $N$  et de tous les arcs (donnés sous forme de couples).  
On verra plus loin que cette forme est d'un usage peu pratique en informatique.
- ❷ On utilise la matrice  $M \in \mathcal{M}_N(\mathbb{R})$  définie par  $M[i, j] = 1$  si  $(i, j) \in A$  et  $M[i, j] = 0$  sinon.  
C'est la **matrice d'adjacence** du graphe. Elle est symétrique s'il est non-orienté.
- ❸ On prend la liste  $L$  de longueur  $N$  telle que  $L[i]$  soit la liste des successeurs de  $i$ .  
C'est la **liste d'adjacence** du graphe.

**Remarque** : le mode de représentation sera le même que le graphe soit orienté ou non.

**Exemple** : considérons le graphe ci-dessous



On peut le représenter des trois manières suivantes :

$$\textcircled{1} G = [6, (0, 1), (0, 2), (1, 2), (2, 3), (4, 5), (5, 4)]$$

$$\textcircled{2} M = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{pmatrix}$$

$$\textcircled{3} L = [[1, 2], [2], [3], [], [5], [4]]$$

On va créer quelques fonctions qui permettent de manipuler ces représentations sous forme de listes ou de matrices d'adjacence :

- `nouveau(N)` qui crée un graphe à  $N$  sommets sans arête
- `ajoute_arete(G, i, j)` qui ajoute l'arête  $(i, j)$  au graphe  $G$
- `supprime_arete(G, i, j)` qui supprime l'arête  $(i, j)$  du graphe  $G$
- `ajoute_sommet(G)` qui ajoute un sommet appelé  $N$  au graphe  $G$
- `supprime_sommet(G, i)` qui supprime le sommet  $i$  et renomme les sommets suivants

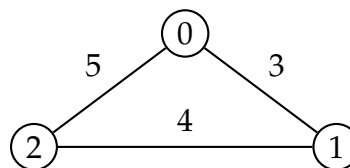
Grâce à ces fonctions, on peut facilement créer un graphe en partant de la liste de ses arêtes et s'amuser à le modifier.

### 3.2) Cas des graphes pondérés

On modifie légèrement chaque type de représentation pour inclure une information supplémentaire, la pondération.

- ① On remplace les couples  $(i, j)$  par des triplets  $(i, j, p)$ , où  $p$  est le poids de l'arc  $(i, j)$ .
- ② On attribue à  $M[i, j]$  la valeur  $p$  si l'arc  $(i, j)$  est de poids  $p$ , et une valeur particulière (comme `None` ou `False`) qui modélise l'infini lorsque l'arc n'existe pas.
- ③  $L[i]$  est la liste des couples  $(j, p)$  tels qu'il existe un arc de poids  $p$  reliant  $i$  à  $j$ .

**Exemple :** considérons le graphe non-orienté pondéré suivant



Voici les trois représentations possibles :

$$\textcircled{1} G = [3, (0, 1, 3), (1, 0, 3), (0, 2, 5), (2, 0, 5), (1, 2, 4), (2, 1, 4)]$$

$$\textcircled{2} M = \begin{pmatrix} \text{None} & 3 & 5 \\ 3 & \text{None} & 4 \\ 5 & 4 & \text{None} \end{pmatrix}$$

$$\textcircled{3} L = [[(1, 3), (2, 5)], [(0, 3), (2, 4)], [(0, 5), (1, 4)]]$$

**Remarque :** en Python, on utilisera `numpy.inf` pour modéliser l'infini dans les matrices d'adjacence car cette quantité se prête aux calculs et aux comparaisons.

### 3.3) Avantages et inconvénients des différentes représentations

- ❶ La représentation par liste est facile à établir par lecture et permet de réaliser rapidement un dessin. Mais elle est peu pratique à l'usage : ainsi, pour savoir si deux sommets sont reliés par une arête, il faut parcourir systématiquement toute la liste des arêtes afin d'extraire celles qui quittent ou rejoignent un sommet fixé. On lui préférera les deux autres types de représentation.
- ❷ La matrice d'adjacence est très simple à utiliser dans un programme informatique, mais elle occupe une place en mémoire de l'ordre de  $N^2$  et ainsi peut s'avérer lente à exploiter dès que  $N$  est grand.
- ❸ La liste d'adjacence n'occupe en mémoire qu'une place de l'ordre de  $\text{card}(A)$  et se prête bien aux algorithmes de parcours, car on a directement les successeurs d'un sommet donné. En contrepartie, elle est un peu moins simple à manipuler que la matrice d'adjacence.

Pour résumer, si l'on s'intéresse aux coûts des stockages et des différentes opérations :

- Les listes d'adjacence occupent moins de place en mémoire pour les graphes peu denses, c'est-à-dire avec un faible nombre d'arêtes. C'est le cas de la plupart des graphes.
- La recherche de l'existence d'un arc est plus rapide avec la matrice d'adjacence.
- Le parcours des arêtes issues d'un sommet est plus rapide avec la liste d'adjacence.

### 3.4) Usage des dictionnaires

Lorsque les sommets ne sont pas forcément des entiers, on peut utiliser un dictionnaire pour implémenter la liste d'adjacence du graphe :

- ses clefs sont les noms des sommets ;
- les valeurs associées sont les listes de successeurs.

Dans le cas de graphes pondérés, les valeurs seront des listes de couples (sommet, poids).

**Exemple :** si l'on prend le réseau d'amitié vu en introduction, on obtient ainsi

```
G = {"Baptiste" : ["Antoine", "Clovis"],
      "Antoine" : ["Baptiste", "Noémie", "Alexandre"],
      "Clovis" : ["Baptiste"],
      "Fantine" : ["Alexandre", "Julia"],
      "Morgane" : ["Alexandre", "Noémie"],
      "Julia" : ["Fantine"],
      "Alexandre" : ["Antoine", "Fantine", "Morgane"],
      "Noémie" : ["Antoine", "Morgane"] }
```

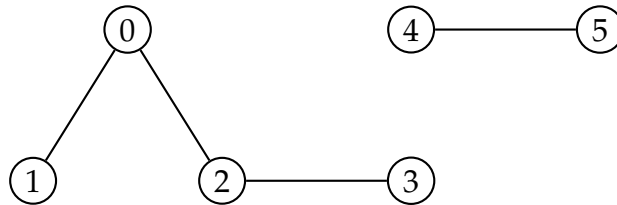
On peut aussi utiliser les dictionnaires afin d'enrichir les informations stockées. Par exemple, les valeurs associées aux sommets peuvent être des dictionnaires contenant leurs degrés entrant et sortant ainsi que la liste de leurs successeurs (sans oublier les poids dans le cas d'un graphe pondéré).

# Exercices

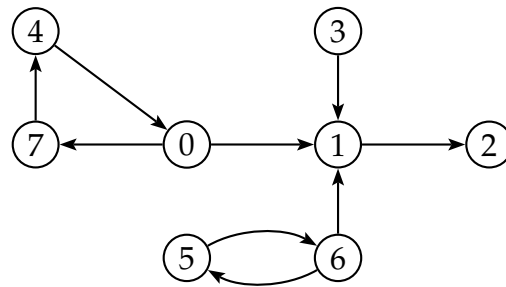
## Exercice 1.

Déterminer les composantes (fortement) connexes des graphes suivants :

1) Graphe n°1 :



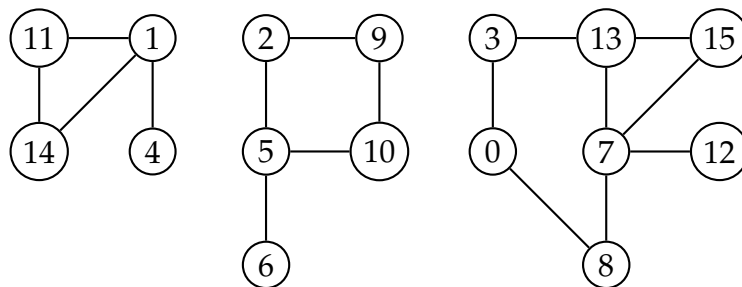
2) Graphe n°2 :



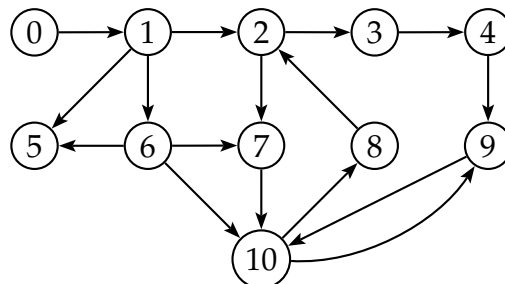
## Exercice 2.

Même question avec les graphes suivants :

1) Graphe n°3 :



2) Graphe n°4 :



**Exercice 3.**

Soit  $G$  un graphe représenté à l'aide de sa liste d'adjacence.

- 1) Écrire une fonction `est_arete(G, i, j)` qui renvoie `True` si  $(i, j)$  est une arête de  $G$  et qui renvoie `False` sinon.
- 2) En déduire une fonction `est_chemin(G, L)` qui renvoie `True` si la liste des éléments de la liste  $L$  représente un chemin dans  $G$  et qui renvoie `False` sinon.

**Exercice 4.**

Soit un graphe défini par son nombre de nœuds et la liste des arêtes les reliant.  
Écrire deux fonctions qui renvoient sa matrice d'adjacence et sa liste d'adjacence.

**Exercice 5.**

On considère un graphe fini.

- 1) Écrire une fonction qui renvoie la liste d'adjacence en partant de la matrice d'adjacence.
- 2) Écrire une fonction qui renvoie la matrice d'adjacence en partant de la liste d'adjacence.

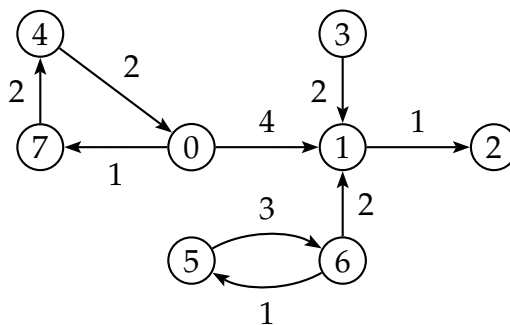
\*\*\* On pourra tester les fonctions des trois exercices précédents sur les graphes n°1 et n°2 \*\*\*

**Exercice 6.**

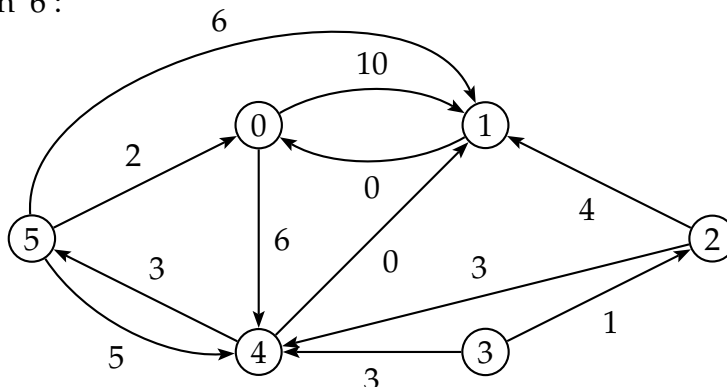
Adapter les exercices 4 et 5 au cas des graphes pondérés.

\*\*\* Pour les tests, on utilisera cette fois les graphes n°5 et n°6 ci-dessous \*\*\*

- Graphe n°5 :



- Graphe n°6 :

**Exercice 7.**

Reprendre ce qui précède en implémentant les listes d'adjacence avec des dictionnaires.