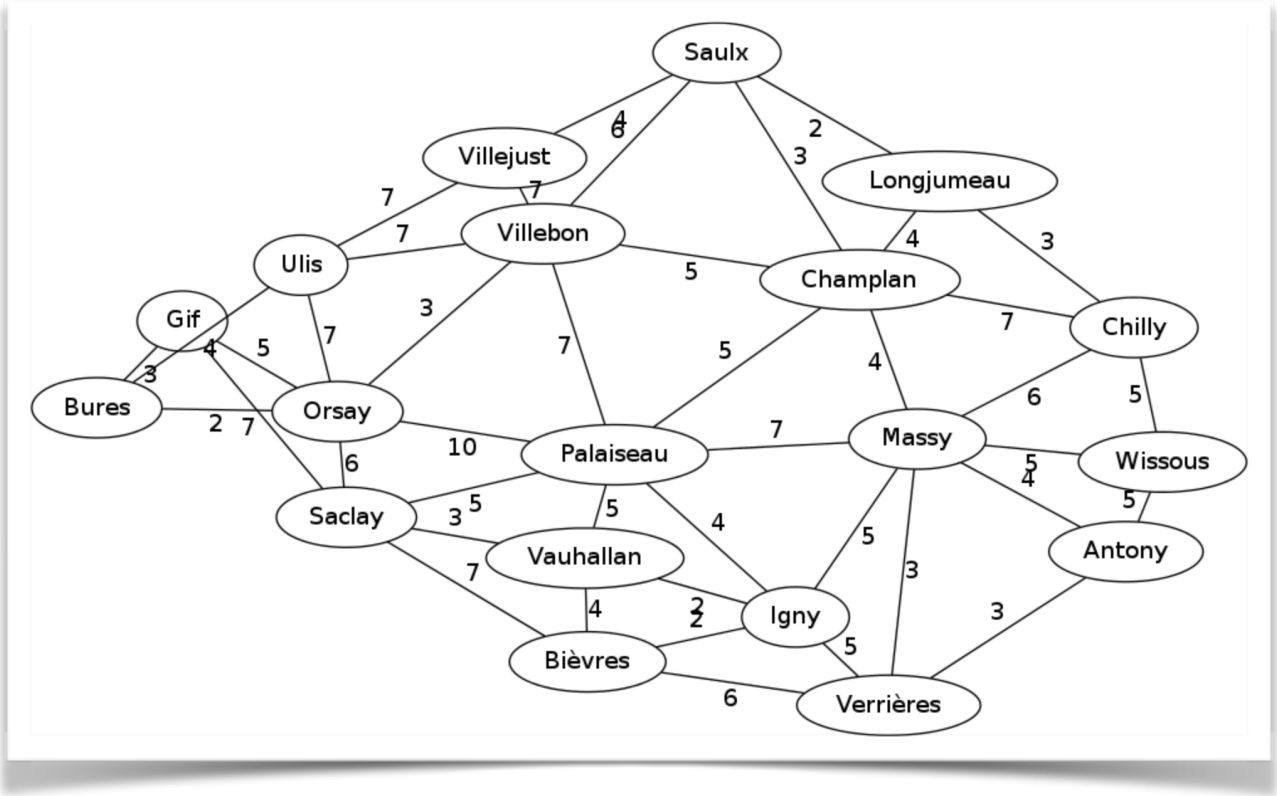
Chapitre 12 : Graphes (1ère partie)

Afin d'illustrer les propos de ce chapitre, nous nous appuierons sur l'exemple suivant représentant une carte simplifiée des alentours de l'école Polytechnique.



1. Qu'est-ce qu'un graphe ?

1.1.Définition mathématique

Un graphe est un objet mathématique constitué :

- D'un ensemble de points noté S
- D'un ensemble de couples (x,y) ou de paires $\{x,y\}$, avec $x \in S$ et $y \in S$, noté A

Ainsi, on note souvent un graphe de la manière suivante : $G(S,A)$.

Il est nécessaire de distinguer paires et couples :

- Un couple (x,y) traduit une relation entre x et y sans que cela implique l'existence d'une relation entre y et x.
- Une paire $\{x,y\}$ traduit une relation symétrique entre x et y. Dans ce casson l'existence d'une relation entre x et y implique nécessairement l'existence d'une relation entre y et x.

Remarque : la principale différence entre un couple et une paire réside dans le fait qu'un couple est ordonné alors qu'une paire ne l'est pas (on peut inverser les rôles de y et x). On parle ainsi de graphe orienté lorsque ce dernier est constitué de couples et de graphe non-orienté s'il s'agit de paires.

1.2. Vocabulaire sur les graphes

- On dit d'un graphe qu'il est constitué de sommets (ou noeuds) et d'arêtes. Les sommets correspondent aux points de l'ensemble S. Les arêtes sont données quant à elle par les éléments de l'ensemble A reliant certains sommets entre eux.
- Lorsque le graphe est orienté, chaque arête possède un sens de parcours ; on parle alors d'arc.
- Il est possible d'attacher des informations aux sommets et aux arcs ; on dit alors du graphe qu'il est étiqueté ou pondéré (voir plus loin).
- On appelle ordre d'un graphe le nombre de ses sommets.
- Deux sommets reliés par une arête sont dits adjacents (ou voisins). Ils constituent les extrémités de l'arête.
- Si le graphe est orienté, on parle d'extrémités initiale (noeud de départ) et finale (noeud d'arrivée).
- Le nombre de voisins d'un sommet s, c'est-à-dire le nombre d'arêtes qui lui sont liées, correspond à son degré noté $d(s)$.
- Si le graphe est orienté, on parle de degré entrant (nombre d'arcs pour lequel le sommet s est l'extrémité finale) et sortant (nombre d'arcs pour lequel le sommet s est l'extrémité initiale), notés respectivement $d_-(s)$ et $d_+(s)$.
- Un sommet de degré 0 est dit isolé.
- Une arête du type $\{x,x\}$ est appelée boucle.
- Un graphe est dit simple dès lors qu'il ne possède pas de boucle et que deux sommets quelconques sont reliés par au plus une arête.
- Un graphe simple est dit complet si tous ses sommets sont adjacents deux à deux.

Dans le cas de notre exemple :
- Les différentes communes constituent les sommets du graphe (ensemble S)
- Les arêtes correspondent aux portions de routes comprises entre deux communes (ensemble A)
- Il s'agit d'un graphe non-orienté (on peut se rendre de Massy à Igny ou inversement d'Igny à Massy)
- Il s'agit d'un graphe simple mais pas d'un graphe complet.
- L'ordre du graphe est 19.
- Le degré du sommet « Palaiseau » est 7

1.3. Chemin, connexité

On peut se déplacer au sein d'un graphe en passant d'un sommet à un sommet adjacent en empruntant l'arête qui les relie.
On appelle chemin menant du sommet dit de départ s_0 au sommet dit d'arrivée s_n, la séquence $(s_0,s_1,...,s_n)$ où deux sommets consécutifs sont adjacents.
Remarque : un chemin peut être également donné par la séquence des arêtes empruntées $(a_1,...,a_n)$.
La longueur d'un chemin correspond au nombre d'arêtes empruntées pour aller du sommet de départ au sommet d'arrivée.
Exemple : (« Villebon », « Saulx », « Longjumeau », « Chilly ») est un chemin menant de « Villebon » à « Chilly ». Sa longueur est 3.

La distance entre deux sommets est la longueur minimale d'un chemin reliant ces deux sommets.
Exemple : La distance de « Villebon » à « Chilly » est 2 (en passant par « Champlan »).

S'il n'existe pas de chemin reliant deux sommets, il existe entre eux une distance infinie.
Si les sommets de départ et d'arrivée d'un chemin sont identiques, alors on parle de cycle.
Exemple : (« Villebon », « Saulx », « Longjumeau », « Chilly », « Champlan », « Villebon »)

Un graphe sera dit connexe si et seulement si, pour tout couple de sommets, il existe au moins un chemin les reliant. C'est le cas du graphe donné en exemple.

1.4. Poids, étiquettes

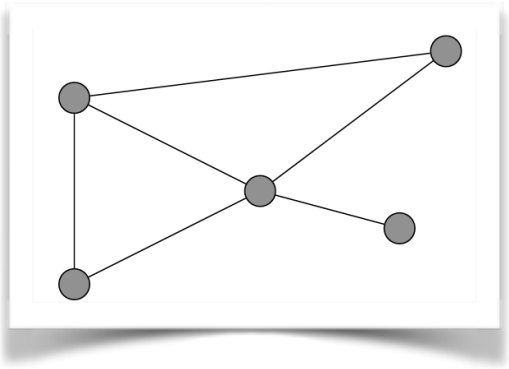
Il est possible d'associer des informations aux arêtes d'un graphe. On parle alors :
- De graphe étiqueté si cette information est un texte (une étiquette)
- De graphe pondéré s'il s'agit d'une information numérique (poids)

Dans notre exemple, il s'agit d'un graphe pondéré puisqu'on a associé à chaque arête la distance entre les communes correspondant aux extrémités.

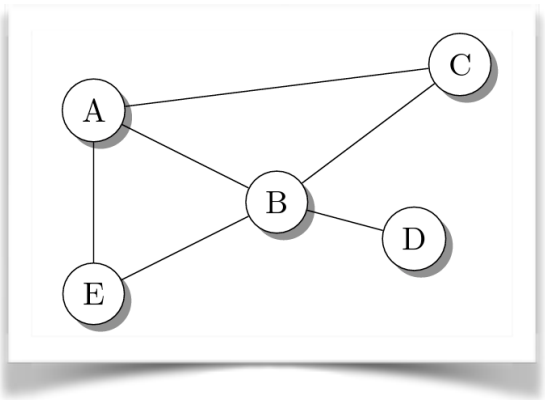
2. Représentations d'un graphe

2.1. Représentation schématique

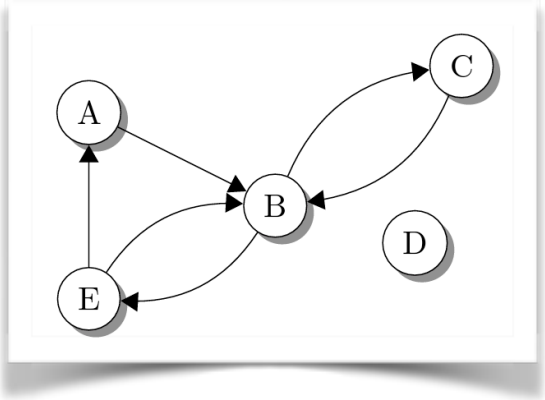
On peut représenter un graphe à l'aide d'un dessin où les sommets sont des points et les les arêtes des lignes reliant deux points.



Il sera souvent plus aisé de nommer les sommets par exemple avec des lettres, même s'il ne faut pas perdre de vue qu'un sommet peut-être une entité complexe telle qu'une personne dans un réseau social, un routeur dans un réseau internet... Voici deux exemples de graphes, l'un non-orienté (graphe 1), l'autre orienté (graphe 2).

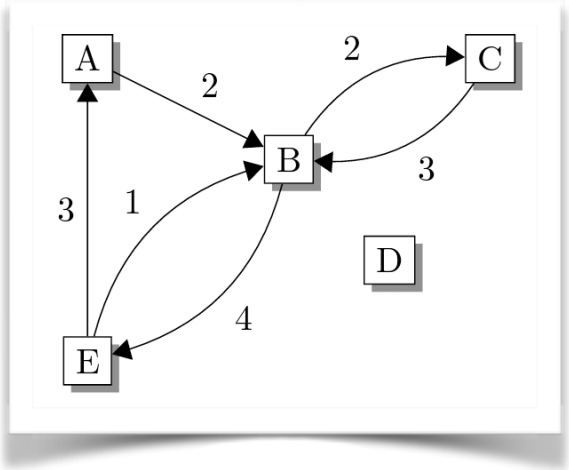


Graphe 1



Graphe 2

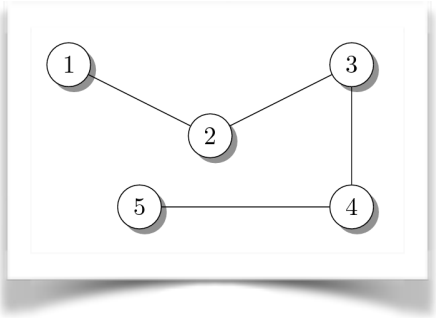
Dans le cas d'un graphe étiqueté ou pondéré, on place les informations le long des arêtes comme dans l'exemple du graphe 3 ci-dessous.



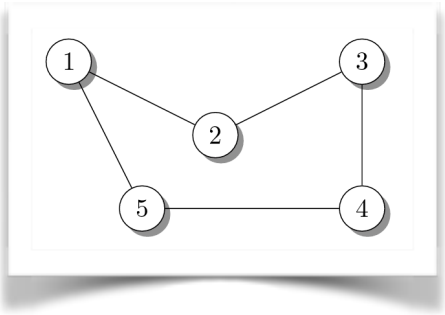
Graphe 3

Quelques cas particuliers à connaître :

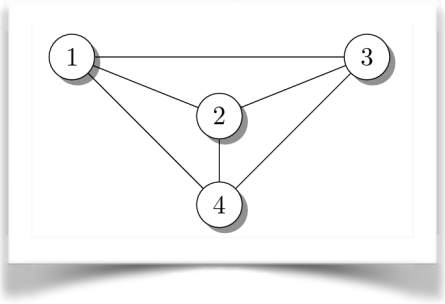
- Graphe de type « chaîne »



- Graphe de type « cycle »



- Graphe complet



2.2. Représentation par des listes d'adjacence

Dans le cas d'un graphe non-orienté, la liste d'adjacence d'un sommet correspond à la liste de ses voisins. L'ordre d'écriture de cette liste n'a pas d'importance.
Pour le graphe 1, on obtient les listes d'adjacence suivantes :
$A : [B, C, E]$
$B : [A, C, D, E]$
$C : [A, B]$
$D : [B]$
$E : [A, B]$

Dans le cas d'un graphe orienté, il peut associer deux listes à un sommet suivant qu'on peut aller vers un voisin (on parlera de successeur) ou en venir (on parlera de prédécesseur). Le donnée d'une des deux listes pour l'ensemble des sommets du graphe est suffisante.
Pour le graphe 2, on obtient les listes de successeurs suivantes :
$A : [B]$
$B : [C, E]$
$C : [B]$
$D : [\ ]$
$E : [A, B]$

Si le graphe est pondéré, alors on vient compléter les sommets figurant dans la liste par les informations. On obtient ainsi pour le graphe 3 :
$A : [(B, 2)]$
$B : [(C, 2), (E, 4)]$
$C : [(B, 3)]$
$D : [\ ]$
$E : [(A, 3), (B, 1)]$

2.3. Représentation par une matrice d'adjacence

Soit un graphe d'ordre n. On peut associer à ce graphe un tableau chaque colonne ou ligne étant associer à un sommet. On viendra y préciser par une valeur numérique, la présence (valeur égale 1) ou l'absence (valeur égale à 0) d'une arête entre les sommets considérés.
Dans le cas du graphe 1, on obtient ainsi :

	A	B	C	D	E
A	0	1	1	0	1
B	1	0	1	1	1
C	1	1	0	0	0
D	0	1	0	0	0
E	1	1	0	0	0

Ce tableau peut être écrit sous forme d'une matrice $n \times n$. On parle de matrice d'adjacence. Pour le graphe 1, on aurait :

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

On peut remarquer que cette matrice ne possède que des 0 sur sa diagonale et qu'il s'agit d'une matrice symétrique. Ceci est lié au fait qu'il s'agit d'un graphe non-orienté.

En effet, si on considère le graphe 2, on obtient la matrice suivante :

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \end{pmatrix}$$

On remarque que la matrice a toujours une diagonale composée de 0 mais n'est plus symétrique ce qui traduit la caractère orienté du graphe.

Dans le cas d'un graphe pondéré, on vient remplacer les 1 par les poids de chaque arête. On obtient ainsi pour le graphe 3 la matrice suivante :

$$\begin{pmatrix} 0 & 2 & 0 & 0 & 0 \\ 0 & 0 & 2 & 0 & 4 \\ 0 & 3 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 3 & 1 & 0 & 0 & 0 \end{pmatrix}$$

3. Implémentation Python

3.1.À l'aide de la matrice d'adjacence

En python, on peut saisir une matrice comme une liste de listes (on parle de sous-listes), chacune de ces sous-listes correspondant à une ligne de la matrice. Les éléments d'une sous-liste correspondent aux n éléments d'une ligne.
Pour le graphe 1, on aurait par exemple :

```
G=[[0,1,1,0,1],[1,0,1,1,1],[1,1,0,0,0],[0,1,0,0,0],[1,1,0,0,0]]
```

3.2.À l'aide des listes d'adjacence

Si on implémente le graphe à l'aide des listes d'adjacence, deux possibilités s'offrent à nous :

- Avec une liste de listes : chaque élément de la liste est une liste contenant un sommet et la liste de ses voisins.
  Ainsi, pour le graphe 1, nous aurions :

```
G=[["A",["B","C","E"]],["B",["A","C","D","E"]],["C",["A","B"]],["D",
["B"]],["E",["A","B"]]]
```

- Avec un dictionnaire : les clés sont les sommets et les valeurs sont les listes de voisins
  Ainsi, pour le graphe 1, nous aurions :
  `G={"A":["B","C","E"],"B":["A","C","D","E"],"C":["A","B"],"D":["B"],"E":["A","B"]}`

Si on considère un graphe pondéré comme le graphe 3, on complète les listes avec des poids.

Exercice : définir le graphe correspondant au graphe 3 à l'aide d'un dictionnaire.

On peut s'interroger sur l'intérêt d'utiliser un dictionnaire plutôt qu'une liste de sous-listes. Supposons que l'on désire connaître les voisins du point `"B"`.

- Dans les cas d'une liste de sous-listes, il va être nécessaire de parcourir la liste jusqu'à identifier `"B"` comme premier élément d'une sous-liste. On pourra alors avoir accès à la liste des voisins.

```
def recherche_voisins(graph,sommet):
    """ graph est un graphe
        sommet est de type string
        renvoie les voisins de sommet"""
    i=0 # initialisation : on considère la première liste d'adjacence
    while graph[i][0]!=sommet: # on parcourt la liste
        i+=1
    return graph[i][1]
```

- Dans le cas d'un dictionnaire, la recherche par clé étant optimisé, il suffit d'écrire `G["B"]` pour que soit afficher la liste des voisins.