T NSI Les Graphes

1. Introduction

Un graphe en informatique est un modèle mathématique qui permet de modéliser un contexte. On les utilise par exemple pour modéliser :

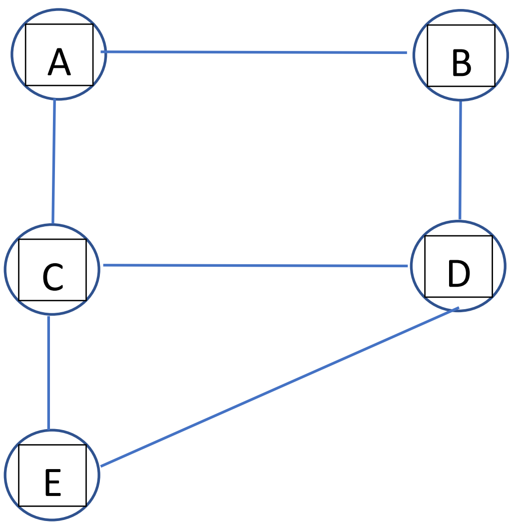
- les relations entre personnes, d'un groupe (réseau sociaux …), un réseau routier ou de métro, un réseau informatique (périphériques terminaux, routeurs ….), les circuits électriques(lois de Kirchhof), une séquence ARN (biologie)

Exemple un réseau social possédant 5 abonnés (A,B,C,D,E):

- A est ami avec B et C
- B est ami avec A et D
- C est ami avec A,E et D
- D est ami avec tout le monde
- E est ami avec C e D

On représente le graphe **non orienté** correspondant par un schéma où

- un **sommet** est un cercle entourant ici le nom de la personne
- une **arête** modélisée par un segment de droite pour la relation d'amitié entre « X » et « Y »



2. Définition cas d'un graphe non orienté

On appelle graphe la donnée d'un ensemble de points appelés **sommets** et d'un ensemble de lignes appelées **arêtes** qui relient certains sommets entre eux.

Un graphe est donné par un couple G = (V, E) où
V est un ensemble (de sommets) ici : V= {A,B,C,D,E }
E est un ensemble de paires {u, v} avec u, v $\in$ V ici ; E = {(A,B),(A,C),(C,D),(B,D),(C,E),(D,E) }

T NSI Les Graphes

Adjacents : deux sommets sont dits adjacents s'ils sont reliés entre eux par une arête.

Adjacentes : deux arêtes sont dites adjacentes si elles possèdent au moins un sommet commun.

Voisins : deux sommets sont dits voisins s'ils sont reliés par une arête

Degré d'un sommet : le nombre d'arêtes dont il est une extrémité c'est aussi le nombre de voisins

L'ordre d'un graphe est son nombre de sommets

Sommet isolé : sommet qui n'est adjacent avec aucun autre

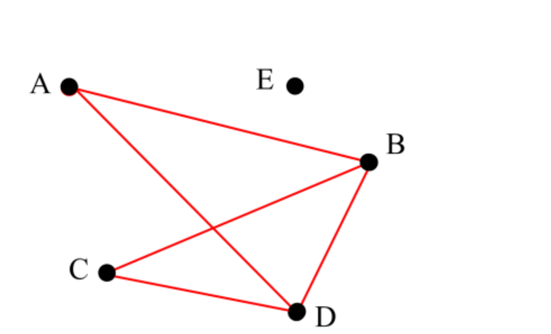
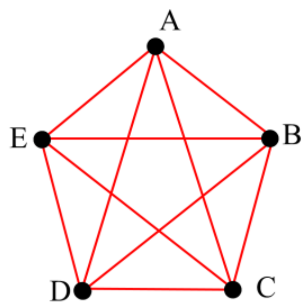
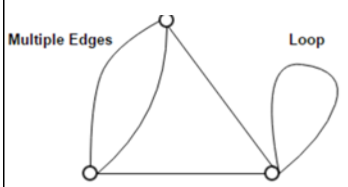
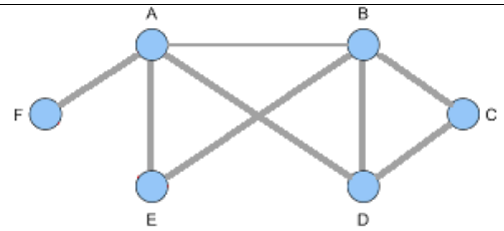
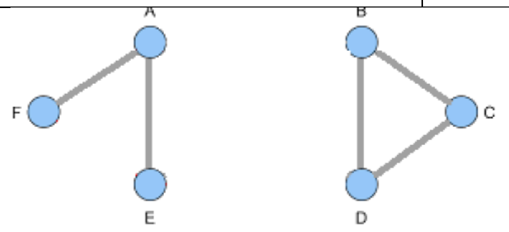
Un graphe simple est un graphe sans boucle tel qu'entre deux sommets il y a au plus une arête.

Un graphe complet est un graphe simple dont tous les sommets sont adjacents avec les autres.

Un graphe connexe est un graphe pour lequel il existe une chaine pour toute pairs de sommets.

Chaîne : dans un graphe non orienté, une chaîne est une suite finie non vide de sommets telle que chaque paire de sommets consécutifs de la suite soit une arête du graphe ; la suite (un moins en longueur) d'arêtes ainsi obtenue caractérise aussi la chaîne. La notion correspondante dans un graphe orienté est celle d'un chemin. La longueur d'une chaîne est celle de sa suite d'arêtes (un moins que la longueur sa suite de sommets). La source de la chaîne est son premier sommet, et son but est son dernier sommet. Une chaîne est dite élémentaire si aucun sommet figure plus d'une fois dans la suite, à l'exception de la source et le but de la chaîne qui peuvent coïncider.

Cycle : chaîne de longueur non nulle dont les sommets de départ et de fin sont les mêmes. (Le cycle est dit simple si les arêtes sont distinctes deux à deux)

<table>
<tr>
<td>
</td>
<td>
</td>
<td>
</td>
</tr>
<tr>
<td>graphe simple d'ordre 5, de sommets A, B, C, D et E. Les sommets A et B sont adjacents, A et C ne le sont pas, E est un sommet isolé.</td>
<td>graphe complet d'ordre 5</td>
<td>Graphe non simple</td>
</tr>
<tr>
<td>
</td>
<td colspan="2">
</td>
</tr>
<tr>
<td>connexe</td>
<td colspan="2">Non connexe</td>
</tr>
</table>

T NSI Les Graphes

Remarque mathématique TH 1 : la somme des degrés de tous les sommets d'un graphe est égale au double du nombre total d'arêtes

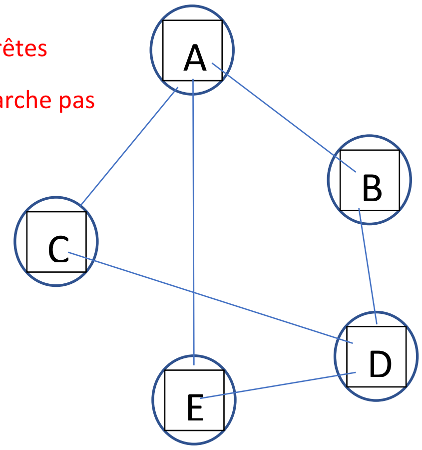
Pour l'exemple : 2+3+2+3+2 = 2 x 6

Application

Cinq joueurs souhaitent organiser un tournoi de tennis ou chaque joueur rencontre 3 autres joueurs. Cela peut-il être organiser ?

Ici somme des degrés 5 x 3 = 2 x arêtes

Arête = 15/é = 7.5 pas entier ne marche pas

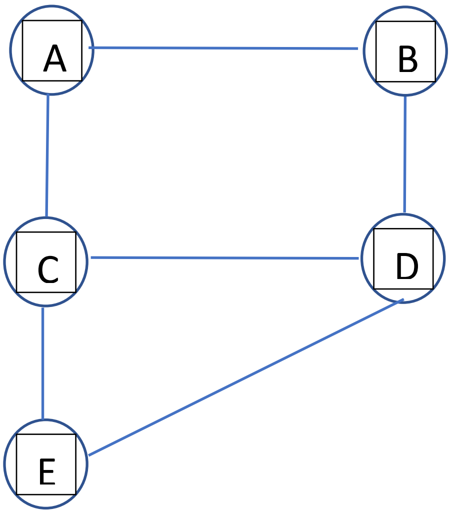


On désire former un graphe avec un hexagone où tous les sommets sont reliés. Combien la figure possède t-elle de segments ?

Nombre de sommets 6 , nombre de degrés 6 soit 6x6 = 2 x arêtes

Arêtes = 36 /2 = 18

Commenter le graphe exemple de départ : Graphe simple d'ordre 5 non complet

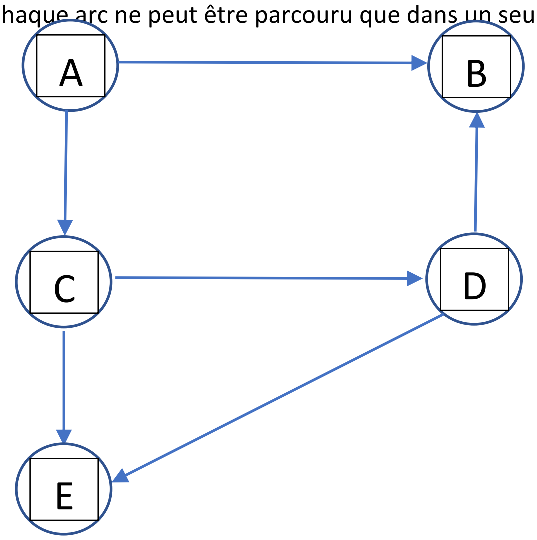


A-C-E-D chaine de longueur 3 A-B-D-E-C-A est un cycle de longueur 5

3. Définition cas d'un graphe orienté

Dans le cas d'un graphe orienté les arêtes s'appellent des **arcs**

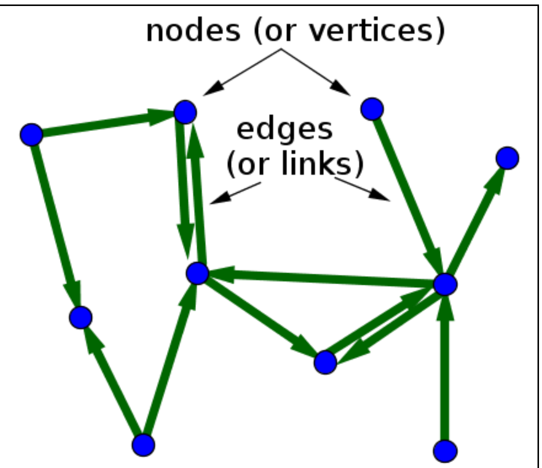
Dans un graphe orienté chaque arc ne peut être parcouru que dans un seul sens indiqué par une flèche



Chemin : dans un graphe orienté, un chemin est une suite finie non vide de sommets telle que chaque couple de sommets consécutifs de la suite soit un arc du graphe ; la suite (un moins en longueur) d'arcs ainsi obtenue caractérise aussi le chemin. La notion correspondante dans un graphe non orienté est celle d'une chaîne. La longueur d'un chemin est celle de sa suite d'arcs (un moins que la longueur sa suite de sommets). La source du chemin est son premier sommet, et son but est son dernier sommet. Un chemin est dit **élémentaire** si aucun des sommets ne figure plus d'une fois comme source, ni plus d'une fois comme but d'un arc du chemin (mais source et but du chemin peuvent coïncider).

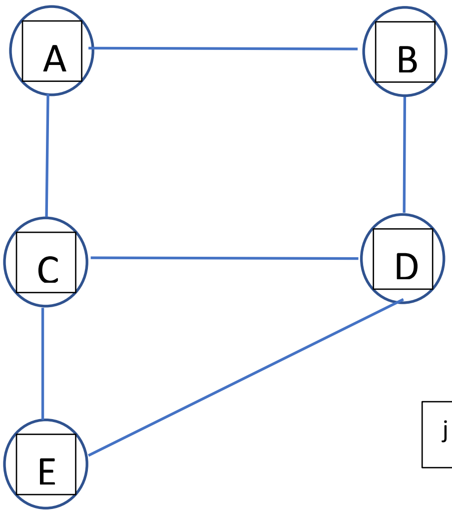
Dans un graphe orienté, on parle d'un **circuit** plutôt que d'un cycle, même si la terminologie n'est pas tout à fait stabilisée.

4. Glossaire franco-anglais

sommet	Vertex /nodes / vertices	
arête	edge	

5. Représentation d’un graphe par une matrice d’adjacence

On peut représenter le graphe grâce à un tableau qui indique l’adjacence entre deux sommets par un 1 et l’absence d’adjacence par un 0. Le tableau est constitué par les colonnes rangeant les sommets dans un certain ordre et des lignes qui range ces mêmes sommets dans le même ordre. C’est ce qu’on appelle une **Matrice d’adjacence**.



i

	A	B	C	D	E
A		**1**	**1**		
B	**1**			**1**	
C	**1**			**1**	**1**
D		**1**	**1**		**1**
E			**1**	**1**	

j

$$A_{i,j} = \begin{cases} 1\ si\ \ (i,j) \in E \\ 0\ sinon \end{cases}$$

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

<u>**Représentation d’une matrice d’adjacence avec python**</u> ici matrice 5 x 5

```
A = [
        [ 0 , 1 , 1 , 0 , 0 ] ,
        [ 1 , 0 , 0 , 1 , 0 ] ,
        [ 1 , 0 , 0 , 1 , 1] ,
        [ 0 , 1 , 1 , 0 , 1] ,
        [ 0 , 0 , 1 , 1 , 0]
    ]
```

<u>**Intérêt de la matrice d’adjacence.**</u>

En **multipliant** une matrice d’adjacence **par n** on peut déterminer le **nombre m** de chaine reliant deux sommets de **longueur n**.

Site https://wims.univ-cotedazur.fr/wims/wims.cgi

$$A^3 = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 0 \end{pmatrix}^3 = \begin{pmatrix} 0 & 4 & 5 & 1 & 2 \\ 4 & 0 & 1 & 5 & 2 \\ 5 & 1 & 2 & 6 & 4 \\ 1 & 5 & 6 & 2 & 4 \\ 2 & 2 & 4 & 4 & 2 \end{pmatrix}$$

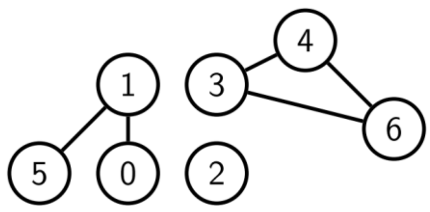
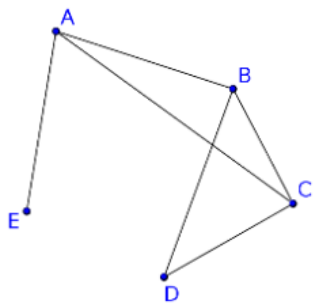
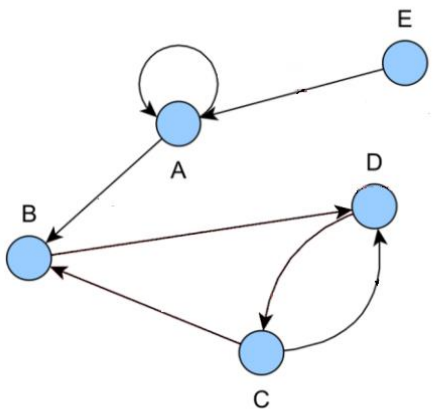
A 2,3

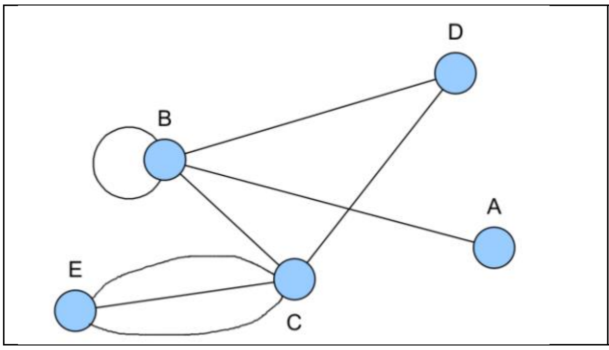
2 sommet C

3 sommet D

Entre les sommets C et D il y a 6 chaines de longueur 3

Donner la matrice d'adjacence des graphes suivant :

	$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$
	$A = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix}$
	$\begin{array}{c c} & \begin{matrix} A & B & C & D & E \end{matrix} \\ \begin{matrix} A \\ B \\ C \\ D \\ E \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$



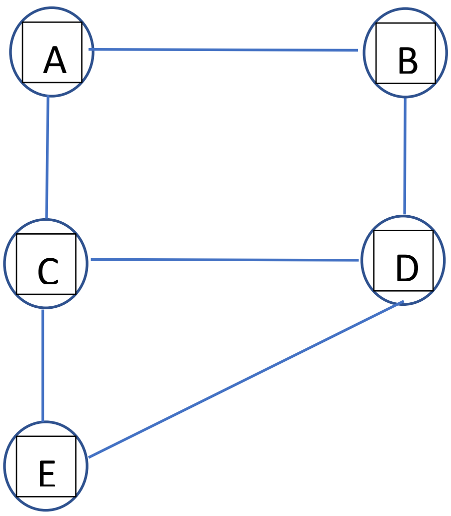
$$\begin{array}{c} \\ A \\ B \\ C \\ D \\ E \end{array}\begin{array}{c} \begin{array}{ccccc} A & B & C & D & E \end{array} \\ \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 3 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 3 & 0 & 0 \end{pmatrix} \end{array}$$

Pour les graphes non orienté la matrice est symétrique (par rapport à la diagonale)

L'inconvénient d'une telle implémentation réside dans l'espace mémoire dédié au graphe. En effet, pour un graphe à n sommets, il faut une matrice avec n^2 nombres… et la plupart du temps, il y a beaucoup de "0" (on parle de *matrices creuses*), ce qui est ballot non ?

6. Représentation d'un graphe avec une liste d'adjacence

Une façon d'encoder un graphe sous Python est d'utiliser un dictionnaire qui sera la représentation de sa matrice d'adjacence. La clé associée à chaque sommet sera la liste des sommets adjacents ou **liste des voisins**.

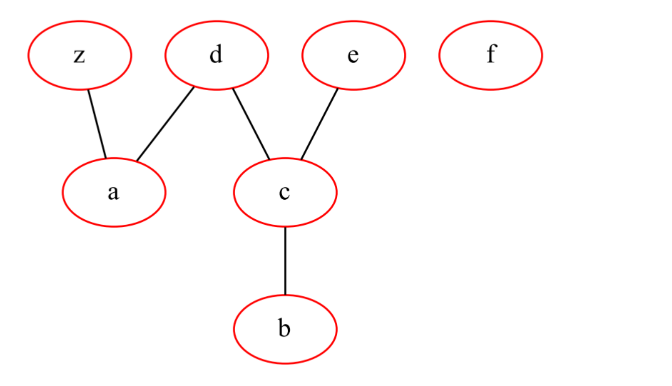



```
g = {    "a" : [ "b" ,"c" ],
         "b" : [ "a", "d"],
         "c" : ["a", "d", "e"],
         "d" : ["b", "c"],
         "e" : ["c", "d"]
}
```

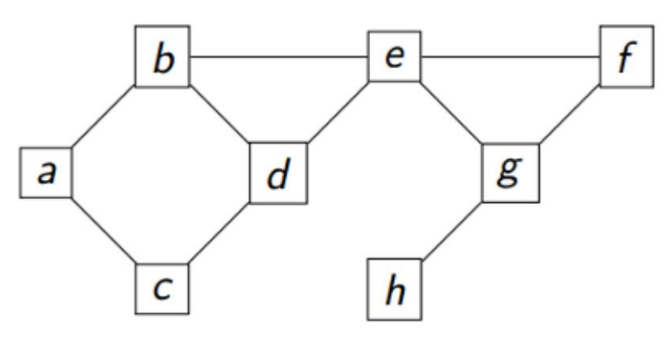
L'implémentation par liste d'adjacence est bien mieux car nécessite moins d'espace mémoire que la représentation par matrice d'adjacence.

Représenter les graphes donnés par les listes d'adjacences :

```
g = {   "a" : ["d"],
        "b" : ["c"],
        "c" : ["b", "d", "e"],
        "d" : ["a", "c"],
        "e" : ["c"],
        "f" : []
}
```

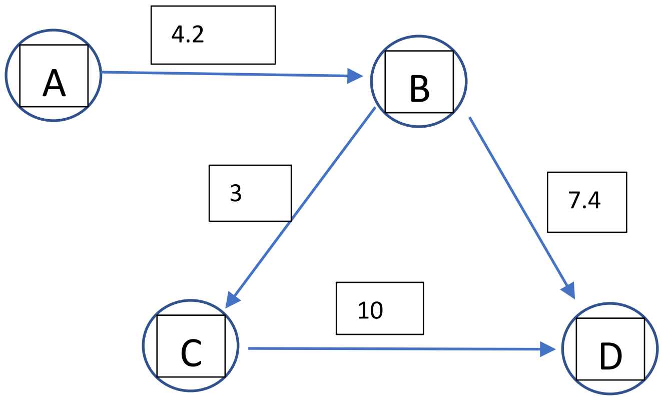



```
G = dict()
G['a'] = ['b','c']
G['b'] = ['a','d','e']
G['c'] = ['a','d']
G['d'] = ['b','c','e']
G['e'] = ['b','d','f','g']
G['f'] = ['e','g']
G['g'] = ['e','f','h']
G['h'] = ['g']
```



7. Cas de graphes pondérés

On appelle un graphe pondéré quand ses arêtes ou ses arcs sont quantifiés par un poids.



Pour la liste d'adjacence on parle ici **de liste des successeurs**.

<table>
<tr><th>Matrice d'adjacence</th><th>Liste d'adjacence</th></tr>
<tr><td>$$A = \begin{pmatrix} 0 & 4,2 & 0 & 0 \\ 0 & 0 & 3 & 7.4 \\ 0 & 0 & 0 & 10 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$</td><td>

```
L = { 'A' : {'B':4.2 },
      'B' : {'C':3,'D':7.4},
      'C' : {'D': 10}
    }
```

</td></tr>
</table>