

M23 : Théorie des Graphes

M. Zimmermann

IUT Robert Schuman, Département Informatique

2022/23

Chapitre 3

Parcours dans un graphe

Remarque :

Avertissement aux lecteurs les plus sensibles : le vocabulaire de ce chapitre est particulièrement peu normalisé.

Des choix ont donc été faits, forcément discutables donc, quant au vocabulaire utilisé (chemins plutôt que chaînes, ...).

Il en va de même en anglais. Il faut donc toujours connaître la définition exacte des auteurs quand ils utilisent ce genre de notions.

Section 1

Se balader dans des graphes

Définition : Chaînes

Soit un graphe $G = (S, A)$ et soit C une séquence de sommets adjacents :

$$C = (s_0, s_1, s_2, \dots, s_k)$$

avec $k > 0$ et s_i des sommets de G , les sommets s_{i-1} et s_i étant adjacents dans G .

- ① C est appelée *chaîne* de s_0 à s_k dans G et k est la *longueur de la chaîne* (c'est le nombre d'arêtes).
- ② Une *chaîne simple* dans G est une chaîne dans G sans répétition d'arête (pas de répétition de deux sommets consécutifs).
- ③ Une *chaîne élémentaire* dans G est une chaîne dans G sans répétition de sommet.

Définition : Chaînes fermées cycles

- ④ Une *chaîne fermée* dans G est une chaîne de G telle que le premier sommet est égal au dernier ($s_0 = s_k$).
- ⑤ Un *cycle* dans G est une chaîne fermée de G simple (pas de répétition d'arête).

On peut parler de cycle élémentaire lorsque que tous les sommets sont distincts (sauf le premier qui est forcément égal au dernier).

Remarques :

- ① Dans le cas des graphes orientés, on préfère le terme « *chemin* » à celui de « chaîne » et on préfère le terme « *circuit* » à celui de « cycle ».
- ② Pour le cas général (en autorisant les graphes orientés, multigraphes...), on pourrait définir un chemin comme une séquence alternée de type :

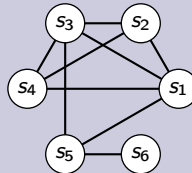
$$C = s_0 a_1 s_1 a_2 s_2 \dots s_{k-1} a_k s_k$$

avec $k > 0$ et s_i des sommets de G , a_i des arêtes de G et $a_i = \{s_{i-1}, s_i\}$ (les sommets s_{i-1} et s_i sont adjacents dans G).

- ③ Une chaîne ou un chemin peuvent aussi être vus comme une suite d'arêtes adjacentes.
- ④ La nomenclature anglaise (tout aussi flottante) utilise des termes comme *walk*, *trek*, *path*, ...

Exemples :

Nous allons nous balader dans le graphe suivant :

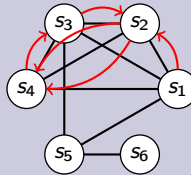


Peut-être pouvez-vous déjà repérer des chaînes, des chaînes fermées, des chaînes simples, des chaînes élémentaires et des cycles.

Exemples :

- ① Considérons la chaîne suivante :

$$C = (s_1, s_2, s_4, s_3, s_2, s_4)$$

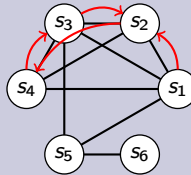


qui n'est ni simple (passe deux fois par l'arête s_2s_4), ni élémentaire (passe deux fois par s_2) ni fermée (le premier sommet est différent du dernier : $s_1 \neq s_4$)

Exemples :

②) Considérons la chaîne suivante :

$$C = (s_1, s_2, s_4, s_3, s_2)$$

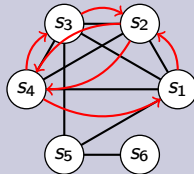


qui est simple (pas de répétition d'arête) mais n'est ni élémentaire (répétition du sommet s_2) ni fermée (le premier sommet est différent du dernier : $s_1 \neq s_2$)

Exemples :

- ③) Considérons la chaîne suivante :

$$C = (s_1, s_2, s_4, s_3, s_2, s_4, s_1)$$

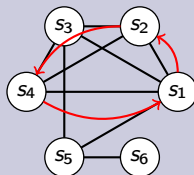


qui n'est ni simple (passe deux fois par l'arête s_2s_4) ni élémentaire (passe deux fois par le sommet s_2) mais est une chaîne fermée (le premier sommet est égal du dernier : $s_1 = s_1$)

Exemples :

- ④) Considérons la chaîne suivante :

$$C = (s_1, s_2, s_4, s_1)$$



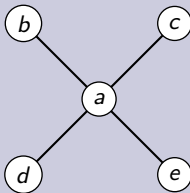
qui est simple (pas de répétition d'arête), et fermée (le premier sommet est égal du dernier : $s_1 = s_1$) donc un cycle qui est élémentaire (pas de répétition de sommet sauf le premier qui est égal au dernier).

Définition : Acyclicité

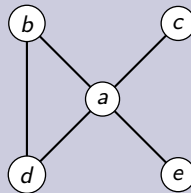
Un graphe est dit *acyclique* lorsqu'il ne contient pas de cycle.

Exemple :

Acyclique



Non acyclique



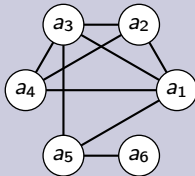
Définition : Connexité

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

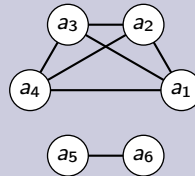
- ① Un graphe est un *graphe connexe* lorsque, pour tout couple de sommets (s, t) de G , il existe une chaîne de s à t .
- ② Un sous-graphe de G est une *composante connexe* de G lorsqu'il est maximal pour la connexité (*i.e.* on ne peut lui ajouter de sommets et/ou d'arêtes supplémentaire sans qu'il ne perde sa propriété de sous-graphe connexe de G)

Exemple :

Connexe



Non connexe



Nb de composantes connexes :

1

2

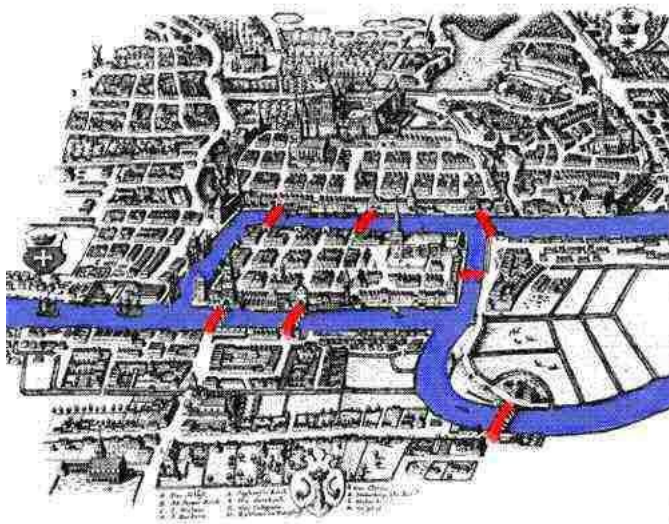
Section 2

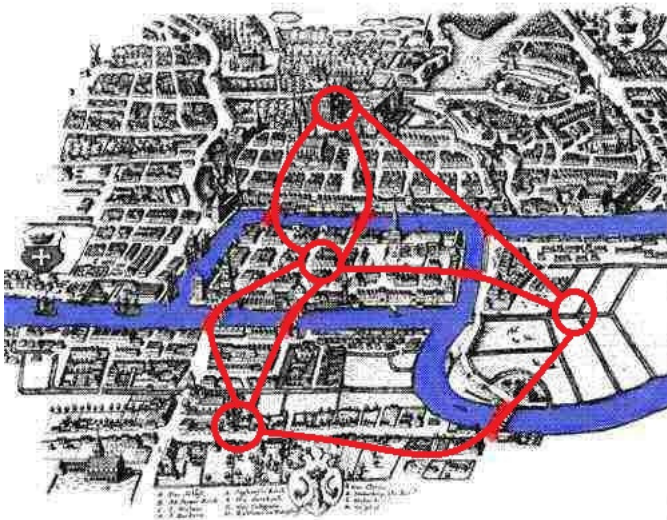
Euler et Hamilton

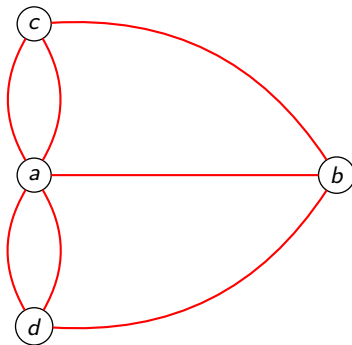
Euler

En 1725, Leonhard Euler (1707-1783) eut vent d'un problème (qu'il qualifia de trivial bien entendu) de balade dans la ville Königsberg (ville natale de Hilbert, E.T.A. Hoffmann et Kant, en Prusse à l'époque, aujourd'hui Kaliningrad en Russie dans un territoire enclavé entre la Pologne et la Lituanie).

Le problème simple à comprendre est un *problème de parcours* : la ville est traversée par la Pregel au-dessus de laquelle étaient construits sept ponts. La question est : peut-on se balader en ville en traversant chaque pont exactement une fois ? Un peu de topologie de la ville :





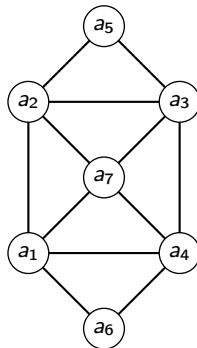


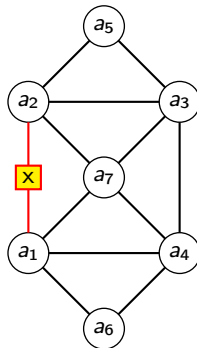
Définition : Graphes eulériens

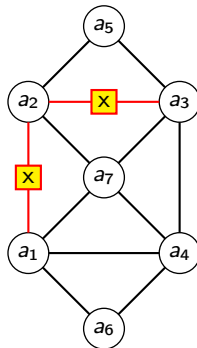
Soit un graphe connexe $G = (S, A)$.

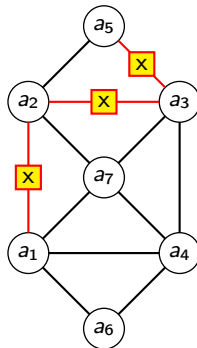
- ① G est appelé *graphe eulérien* lorsqu'il existe un cycle passant par toutes les arêtes. Un tel cycle est appelé *cycle eulérien*.
- ② G est appelé *graphe semi-eulérien* lorsqu'il existe une chaîne passant par toutes les arêtes mais pas de cycle. Une telle chaîne est appelée *chaîne eulérienne*.

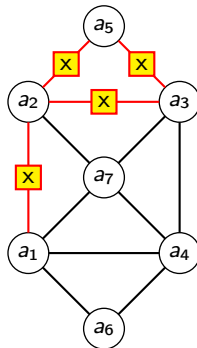
Exemple de cycle eulérien

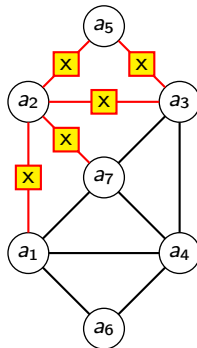


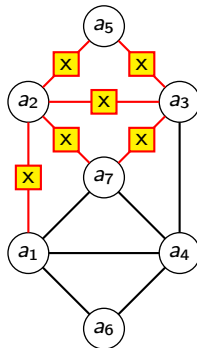


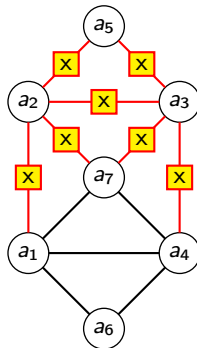


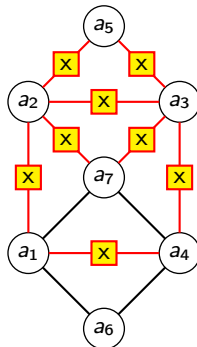


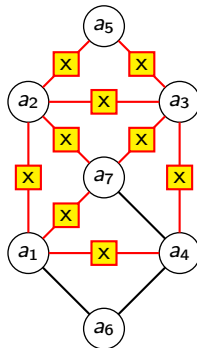


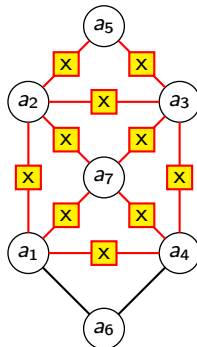


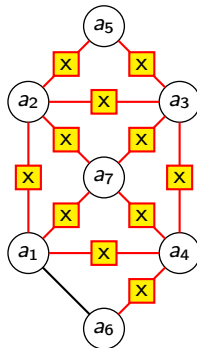


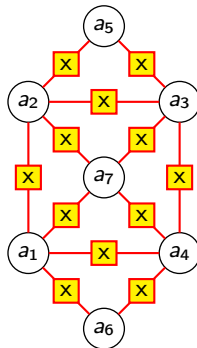




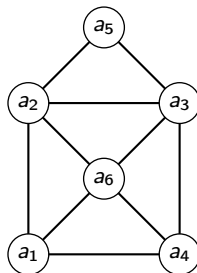


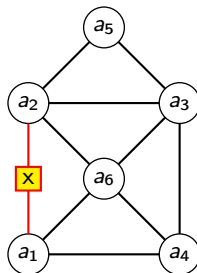


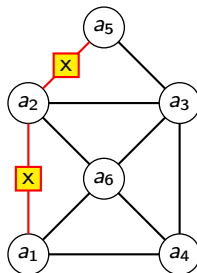


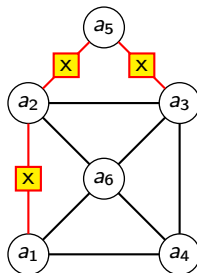


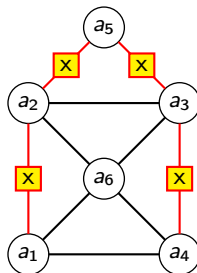
Exemple de chaîne semi-eulérienne

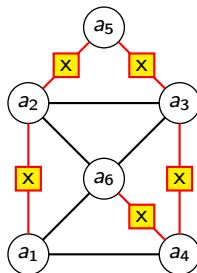


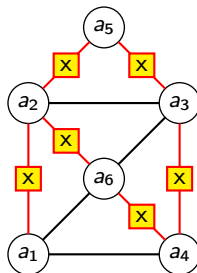


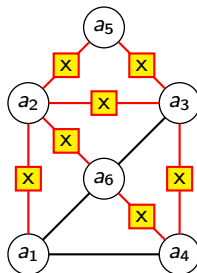


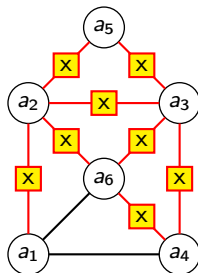


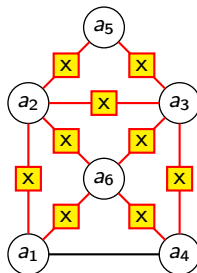


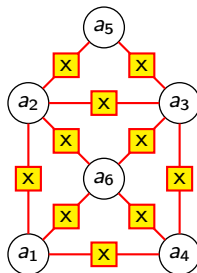












Théorème d'Euler-Hierholzer

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

- ① Pour que G soit un graphe eulérien, il faut et il suffit que chacun de ses sommets soit de degré pair.
- ② Pour que G soit un graphe semi-eulérien, il faut et il suffit que chacun de ses sommets soit de degré pair sauf exactement deux.

Remarque :

Euler fait partie de ces mathématiciens qui ont produit tellement de théorèmes (cf. M11) que pour les différencier, il est préférable d'accoler le nom d'un autre mathématicien pour les différencier. Dans notre cas, il s'agit d'un mathématicien qui a publié une démonstration du résultat.

Nous n'aurons pas d'autre théorème d'Euler dans ce cours, on dira donc tout simplement « théorème d'Euler ».

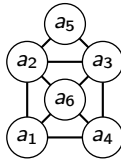
Démonstration : Supposons que le graphe soit eulérien et démontrons que tous ses sommets sont de degré pair.

En parcourant le graphe en utilisant un cycle eulérien, on se rend compte de deux choses :

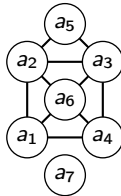
- ① chaque sommet qui n'est pas le premier sommet est un sommet de transit : on y arrive via une arête et on en repart via une autre, c'est-à-dire qu'à chaque passage par un sommet, on augmente son degré de 2. Or on utilise toutes les arêtes, donc le sommet est de degré pair.
- ② Pour le premier sommet, c'est le départ du parcours et l'arrivée du parcours, son degré est donc au moins de 2. Sinon, il peut aussi être un sommet de passage comme précédemment, ce qui nous permet de conclure qu'il est aussi de degré pair car à chaque passage, son degré augmente de 2.

Réciproque : on fera une démonstration algorithmique grâce à un algorithme à venir.

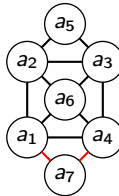
Démonstration : Pour le cas semi-eulérien, il suffit de voir qu'une chaîne semi-eulérienne devient un cycle eulérien dans le graphe obtenu à partir de G en lui ajoutant un sommet et deux arêtes le connectant au sommet de départ et au sommet d'arrivée de la chaîne.



Démonstration : Pour le cas semi-eulérien, il suffit de voir qu'une chaîne semi-eulérienne devient un cycle eulérien dans le graphe obtenu à partir de G en lui ajoutant un sommet et deux arêtes le connectant au sommet de départ et au sommet d'arrivée de la chaîne.



Démonstration : Pour le cas semi-eulérien, il suffit de voir qu'une chaîne semi-eulérienne devient un cycle eulérien dans le graphe obtenu à partir de G en lui ajoutant un sommet et deux arêtes le connectant au sommet de départ et au sommet d'arrivée de la chaîne.



Algorithme : Cycle eulérien

Données : graphe eulérien G eulérien sur les sommets.

Notons s_0 le premier sommet pour cet ordre.

On part de s_0 et on explore les sommets dans l'ordre jusqu'à revenir au premier sommet s_0 sans passer deux fois par la même arête. On obtient ainsi un premier cycle.

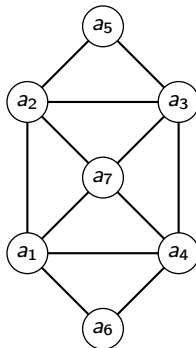
Tant que le cycle ne passe par toutes les arêtes :

On part du premier sommet t rencontré sur le cycle dont on n'a pas encore utilisé toutes les arêtes incidentes et on cherche un cycle comme précédemment mais en partant de t .

On insère le nouveau cycle au précédent pour obtenir un nouveau cycle.

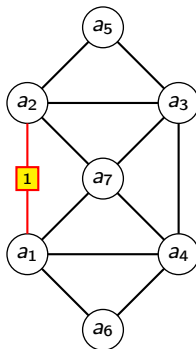
Recherche de parcours eulérien

Graphe de départ ;



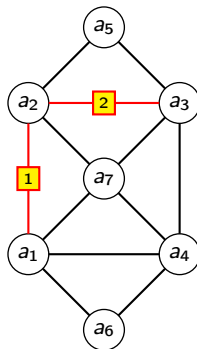
Recherche de parcours eulérien

Exploration :



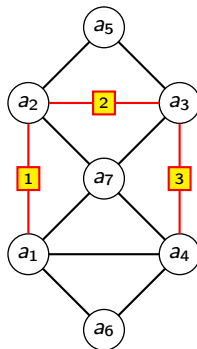
Recherche de parcours eulérien

Exploration :



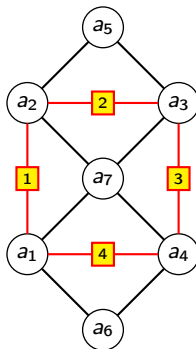
Recherche de parcours eulérien

Exploration :



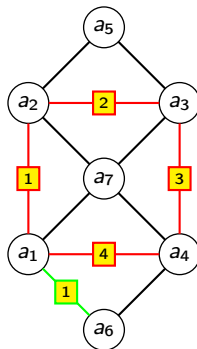
Recherche de parcours eulérien

Exploration :



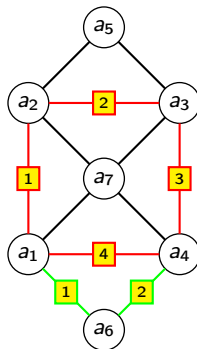
Recherche de parcours eulérien

Exploration :



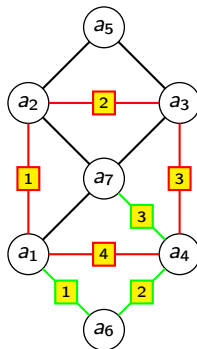
Recherche de parcours eulérien

Exploration :



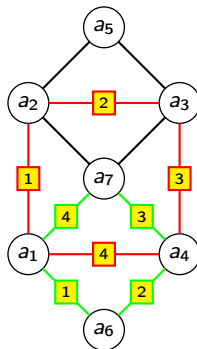
Recherche de parcours eulérien

Exploration :



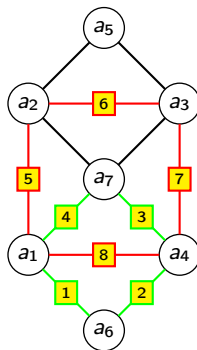
Recherche de parcours eulérien

Exploration :

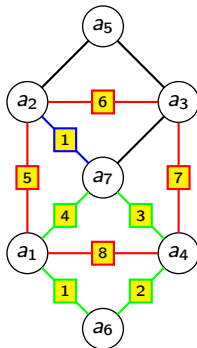


Recherche de parcours eulérien

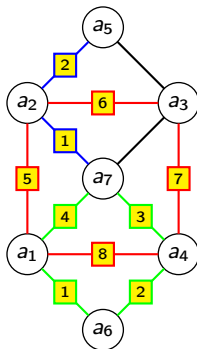
Insertion de la nouvelle chaîne :



Exploration :

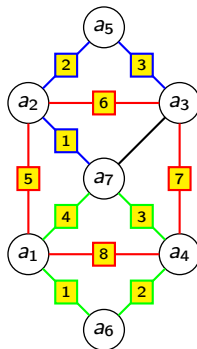


Exploration :

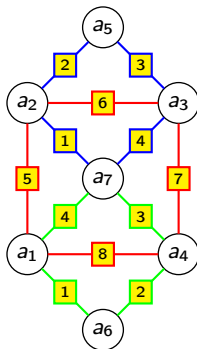


Recherche de parcours eulérien

Exploration :

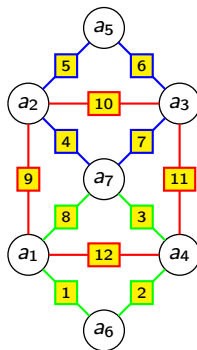


Exploration :



Recherche de parcours eulérien

Insertion de la nouvelle chaîne :



Hamilton

Définition : Graphes hamiltoniens

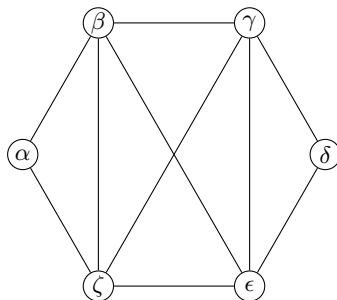
Soit un graphe connexe $G = (S, A)$.

- ① G est appelé *graphe hamiltonien* lorsqu'il existe un chaîne fermée élémentaire passant par tous les sommets. Une telle chaîne est appelé *cycle hamiltonien*.
- ② G est appelé *graphe semi-hamiltonien* lorsqu'il existe une chaîne élémentaire passant par tous les sommets sans pouvoir se refermer en un cycle. Une telle chaîne est appelée *chaîne hamiltonienne*.

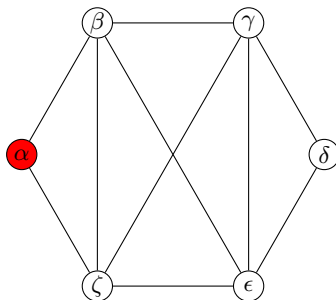
Remarques :

- ① C'est un problème proche de celui d'Euler. Dans le problème eulérien, on cherche à parcourir le graphe en passant exactement une fois par chaque arête (chaînes simples). Le problème hamiltonien consiste à parcourir le graphe en passant exactement une fois par chaque sommet (chaîne élémentaire).
- ② Il n'existe malheureusement pas de théorème « d'Hamilton - Hierzholer ». . . Alors que le problème eulérien est trivial, le problème hamiltonien est compliqué et est même un problème NP-complet.

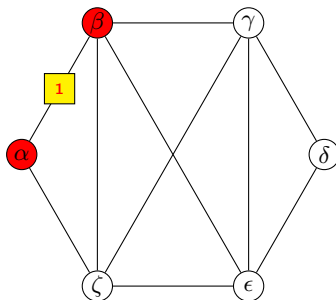
Cycle hamiltonien



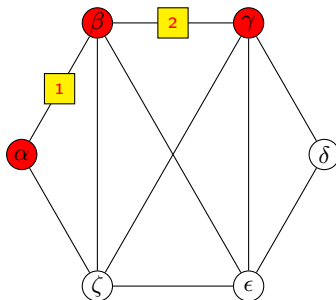
Cycle hamiltonien



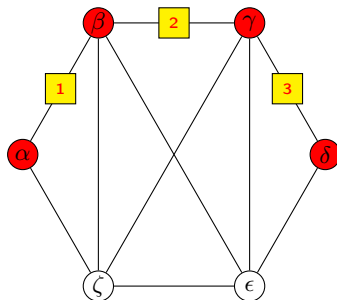
Cycle hamiltonien



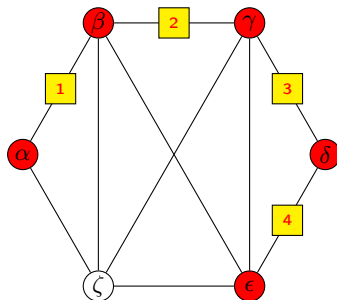
Cycle hamiltonien



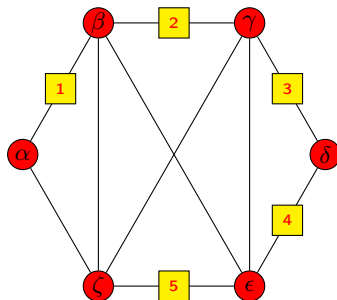
Cycle hamiltonien



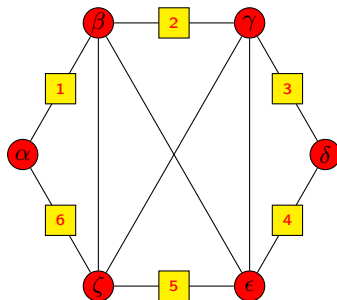
Cycle hamiltonien



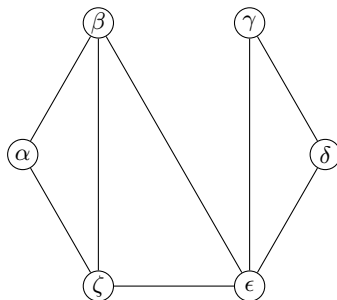
Cycle hamiltonien



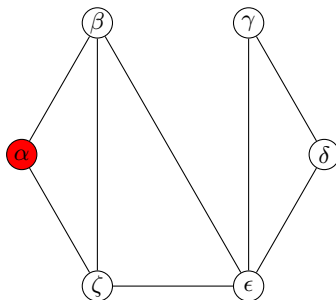
Cycle hamiltonien



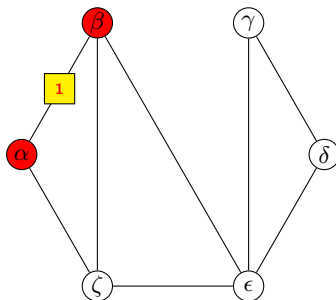
Chaîne (semi-)hamiltonienne



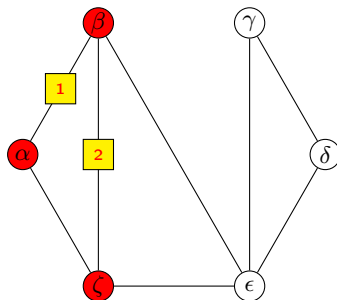
Chaîne (semi-)hamiltonienne



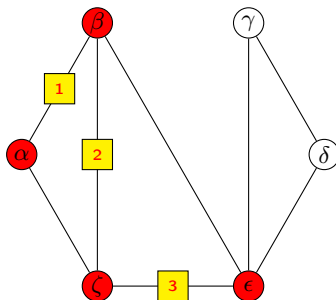
Chaîne (semi-)hamiltonienne



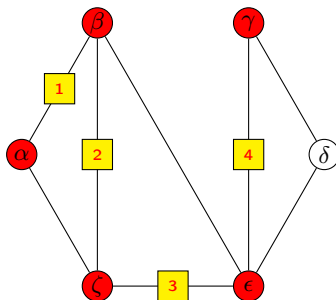
Chaîne (semi-)hamiltonienne



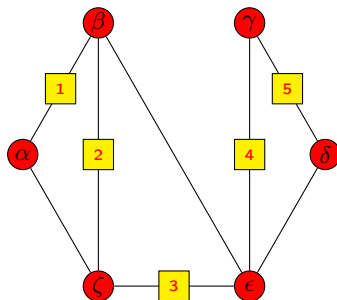
Chaîne (semi-)hamiltonienne



Chaîne (semi-)hamiltonienne



Chaîne (semi-)hamiltonienne



Section 3

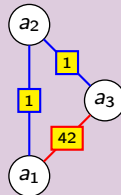
Distances

Définition : Distances

- ① Dans un graphe, la *distance entre deux sommets* est la longueur de la plus petite chaîne les reliant. On notera cette distance $\delta(s, t)$ (avec un δ comme δ istance).
- ② Dans un graphe valué dont les poids sont réels positifs, la *distance pondérée entre deux sommets* (ou *distance valuée* est le plus petit poids total des chaînes reliant les deux sommets. On notera cette distance $d(s, t)$.

Remarques :

- ① La distance peut être vue comme une distance pondérée avec tous les poids égaux à 1.
- ② Les chaînes donnant la distance et la distance pondérée ne sont pas forcément les mêmes :



$\delta(a_1, a_3) = 1$ via la chaîne (a_1, a_3) .

$d(a_1, a_3) = 2$ via la chaîne (a_1, a_2, a_3) .

L'algorithme de Dijkstra est un algorithme très connu pour calculer les distances pondérées d'un sommet fixé s à tous les autres sommets du graphe.

Une information importante :

<https://www.youtube.com/watch?v=yGgWmF6SAPQ>

Algorithme : Dijkstra

Entrées : un graphe connexe à valuation réelle positive G et un sommet $s \in G$.

Initialisation : on initialise le tableau des prédécesseurs $pred$ à *null*, celui des distances à s $dist$ à $+\infty$ et celui les sommets fixés fix à *Faux*, tous de taille G .

$dist[s] = 0$, $pred[s] = s$, $fix[s] = Vrai$, $u = s$

Tant que tous les sommets ne sont pas fixés :

 Pour tous les sommets v de G adjacents à u faire :

 Si $val(u, v) + dist[u] < dist[v]$ alors :

$dist[v] \leftarrow val(u, v) + dist[u]$

$pred[v] \leftarrow u$

 Fin de si

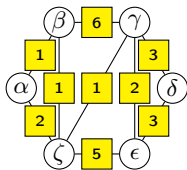
 Fin de pour

$u \leftarrow$ premier sommet non fixé de plus petite distance à s

Fin de tant que

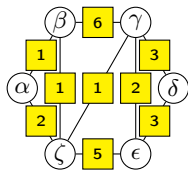
Sortie : Distances de tous les sommets à s .

Algorithme de Dijkstra



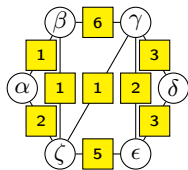
α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α

Algorithme de Dijkstra



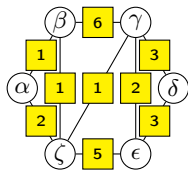
α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α
	1_α	∞	∞	∞	2_α	β

Algorithme de Dijkstra



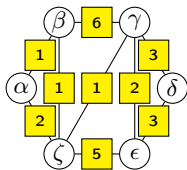
α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α
	1_α	∞	∞	∞	2_α	β
		7_β	∞	∞	2_α	ζ

Algorithme de Dijkstra



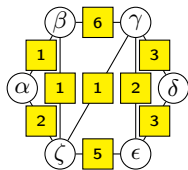
α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α
	1_α	∞	∞	∞	2_α	β
		7_β	∞	∞	2_α	ζ
		3_ζ	∞	7_ζ		γ

Algorithme de Dijkstra



α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α
	1_α	∞	∞	∞	2_α	β
		7_β	∞	∞	2_α	ζ
		3_ζ	∞	7_ζ		γ
			6_γ	5_γ		ϵ

Algorithme de Dijkstra



α	β	γ	δ	ϵ	ζ	fixé
0_α	∞	∞	∞	∞	∞	α
	1_α	∞	∞	∞	2_α	β
		7_β	∞	∞	2_α	ζ
		3_ζ	∞	7_ζ		γ
			6_γ	5_γ		ϵ
			6_γ			δ