THÉORIE DES GRAPHES (2)

Michel Rigo

http://www.discmath.ulg.ac.be/

Année 2007–2008

UNIVERSITÉ de Liège

CHEMINS

DÉFINITION

Soit $G = (V, E)$ un multi-graphe <u>non orienté</u>.

Un chemin de longueur $k \geq 1$ est une suite ordonnée $(e_1, \ldots, e_k)$ de k arêtes adjacentes $e_i = \{e_{i,1}, e_{i,2}\}$,

$$\forall i \in \{1, \ldots, k-1\}, \quad e_{i,2} = e_{i+1,1}.$$

Ce chemin de longueur k joint les sommets $e_{1,1}$ et $e_{k,2}$, passe par les arêtes $e_1, \ldots, e_k$, les sommets $e_{1,1}, e_{1,2}, \ldots, e_{k,1}, e_{k,2}$.

Un chemin de longueur 0 joint toujours un sommet à lui-même.

Si $e_{1,1} = e_{k,2}$: cycle, circuit, chemin fermé

CHEMINS

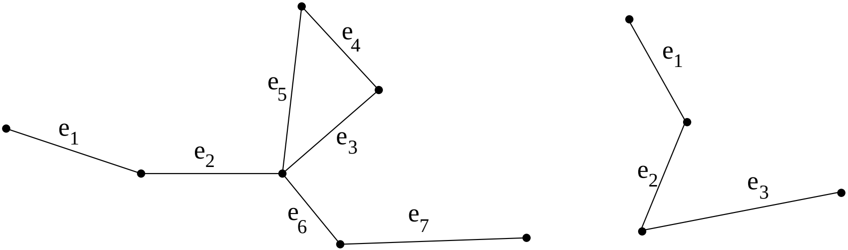
DÉFINITIONS (SUITE)

arêtes d'un chemin toutes distinctes : piste ou chemin élémentaire

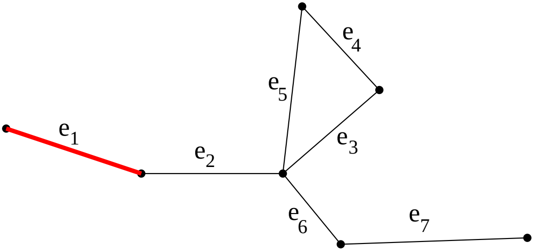
arêtes d'un chemin toutes distinctes et si le chemin ne passe pas deux fois par un même sommet : chemin simple

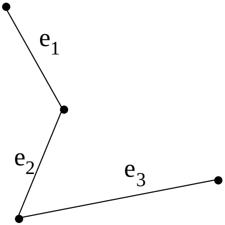
idem, circuit élémentaire / simple

PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE

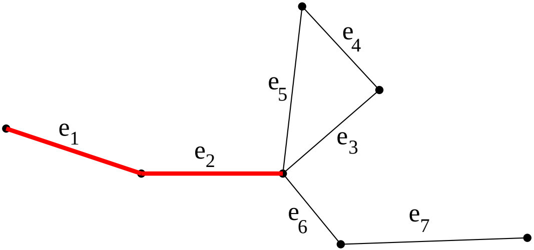


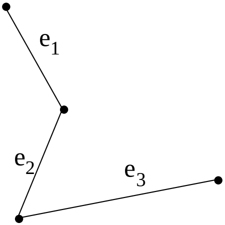
Piste (ou chemin élémentaire) — chemin simple



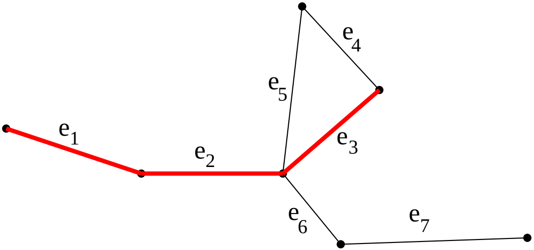


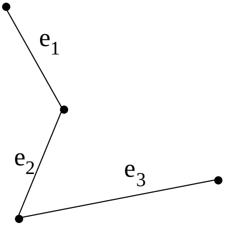
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



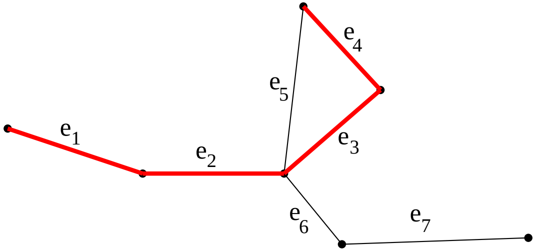


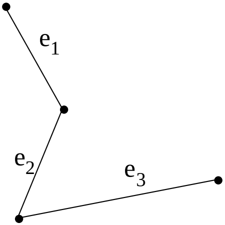
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



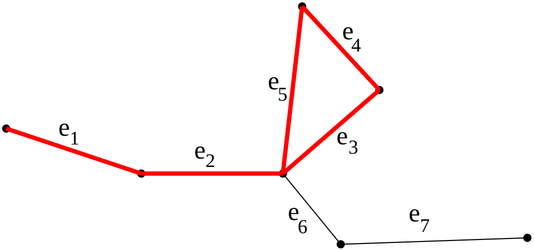


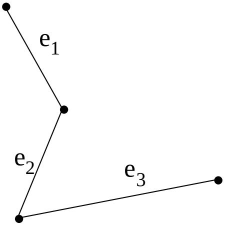
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



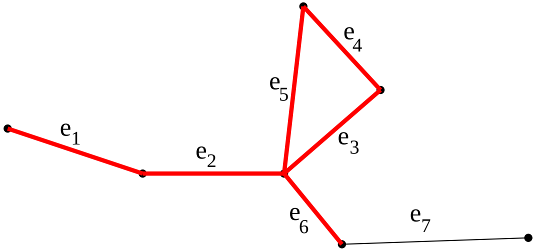


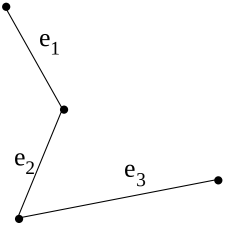
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



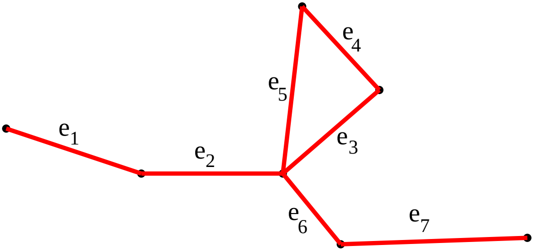


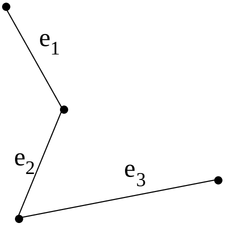
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



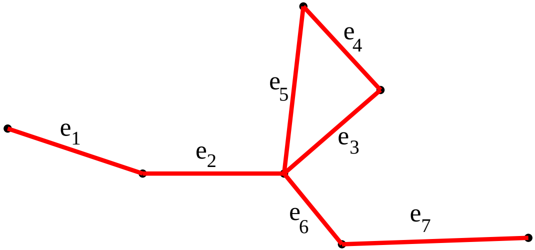


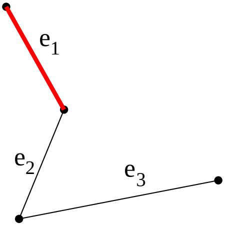
Piste (ou chemin élémentaire) — chemin simple



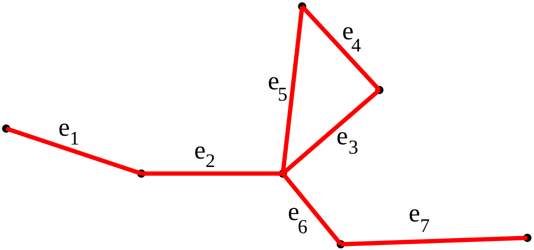


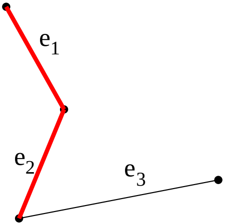
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



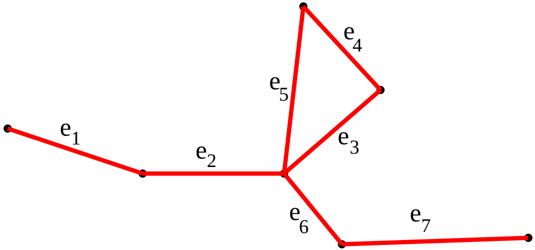


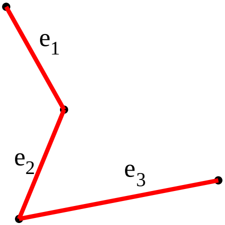
PISTE (OU CHEMIN ÉLÉMENTAIRE) — CHEMIN SIMPLE



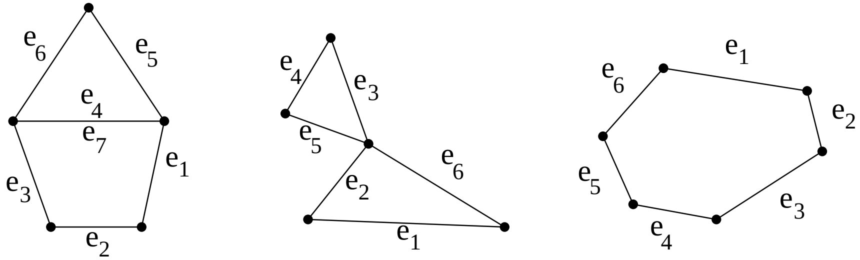


Piste (ou chemin élémentaire) — chemin simple

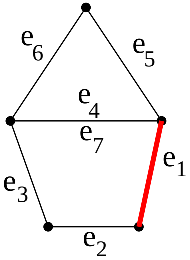


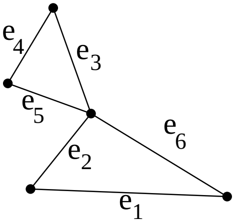


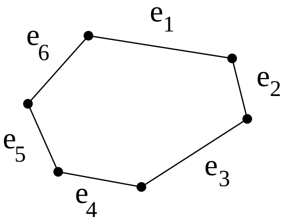
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



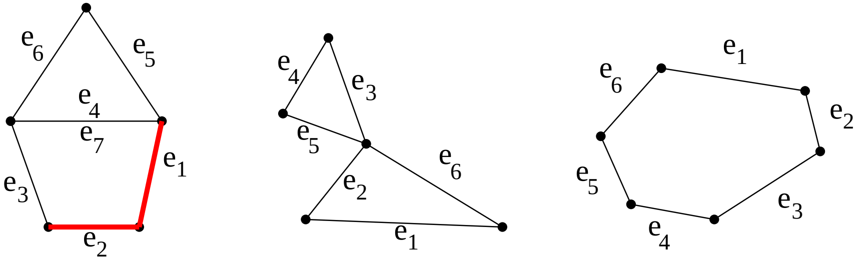
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



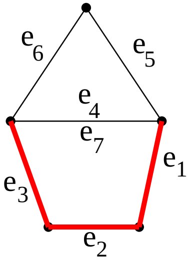


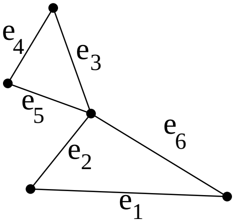


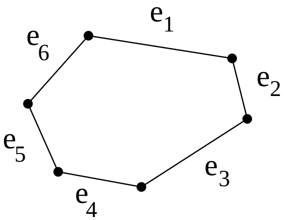
Circuit, circuit élémentaire (ou piste fermée) et circuit simple



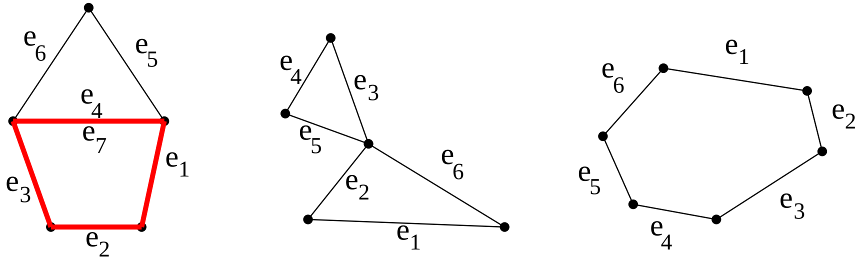
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



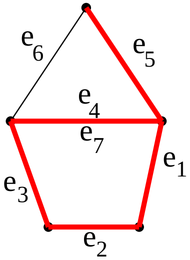


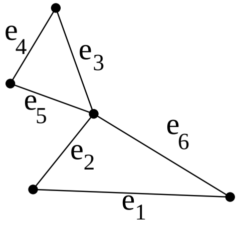


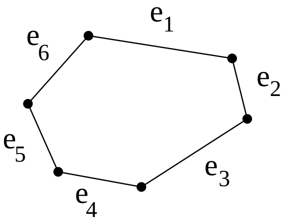
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



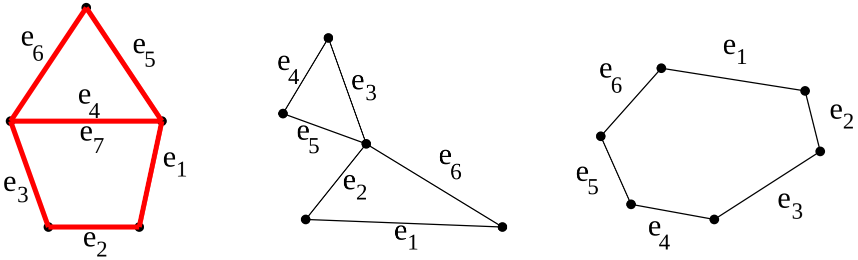
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



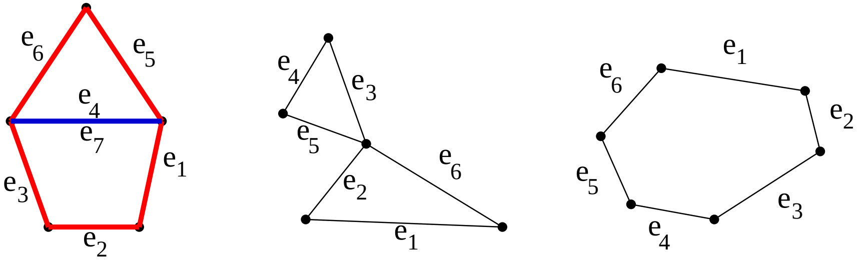




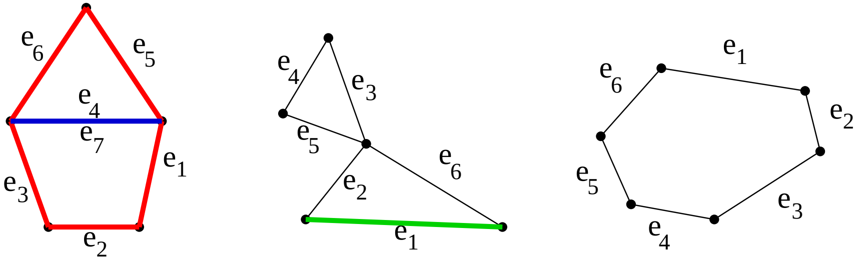
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



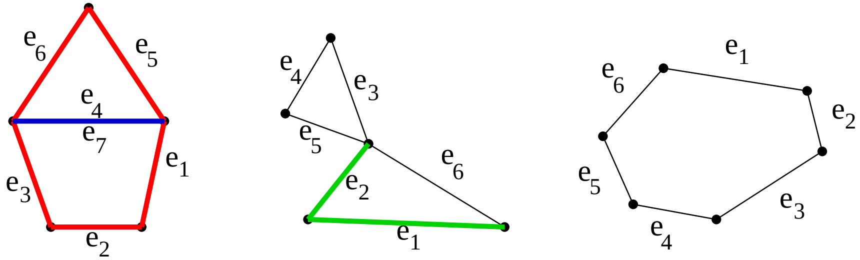
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



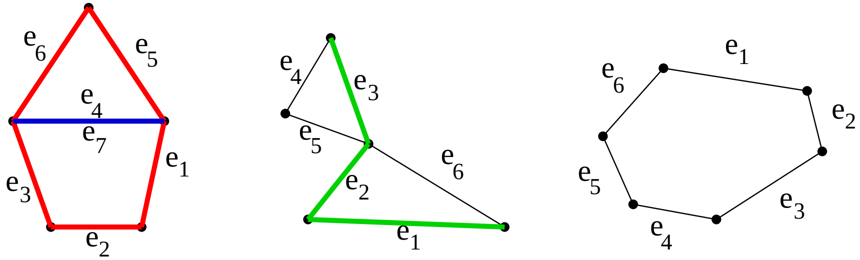
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



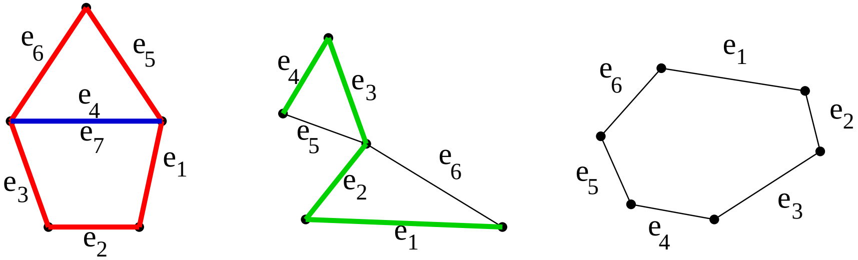
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



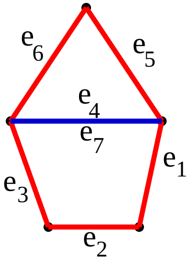
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE

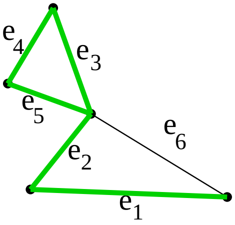


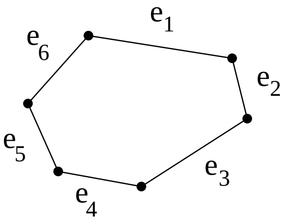
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



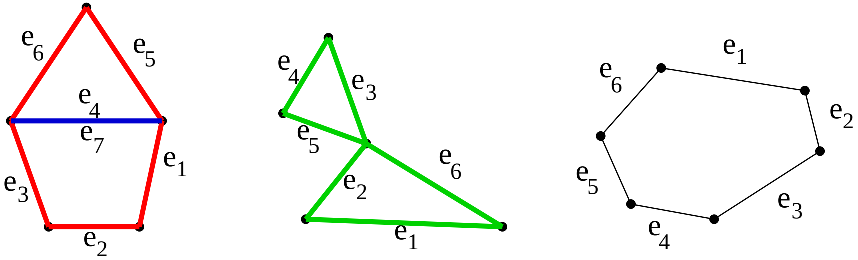
CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE







CIRCUIT, CIRCUIT ÉLÉMENTAIRE (OU PISTE FERMÉE) ET CIRCUIT SIMPLE



Remarque

Dans le cas d'un graphe (qui n'est pas un multi-graphe), un chemin est aussi univoquement déterminé par une suite de sommets $(v_1, \ldots, v_k)$ de manière telle que $\{v_i, v_{i+1}\}$ est une arête du graphe.

Relation d'équivalence

Définition

Deux sommets a et b sont connectés s'il existe un chemin les joignant : $a \sim b$

classe d'équivalence pour $\sim$ = composante connexe

Un multi-graphe non orienté est connexe si $V/\!\sim$ contient une seule composante connexe.

On supposera que $G = (\{v\}, \emptyset)$ est connexe

DÉFINTION

Soit $G = (V, E)$ un multi-graphe non orienté connexe (pour avoir une fonction totale). La distance $\mathrm{d}(a, b)$ entre a et $b \in V$ est la longueur du plus court chemin joignant a et b.

$$\mathrm{diam}(G) = \max_{a,b \in V} \mathrm{d}(a, b).$$

Si G est pondéré par $f : E \to \mathbb{R}^+$, la distance entre les sommets a et b est égale au poids minimal des chemins joignant a et b, i.e.,

$$\mathrm{d}(a, b) = \min_{\substack{\text{chemin } (e_1,\ldots,e_t) \\ \text{joignant } a \text{ et } b}} \sum_{i=1}^{t} f(e_i).$$

DÉFINITIONS

$G = (V, E)$ multi-graphe orienté.
chemin de longueur $k \geq 1$: suite ordonnée

$$(v_1, \ldots, v_k)$$

de k arcs $v_i = (v_{i,1}, v_{i,2})$ t.q. $\forall i \in \{1, \ldots, k-1\}$, $v_{i,2} = v_{i+1,1}$.

Ce chemin de longueur k joint les sommets $v_{1,1}$ et $v_{k,2}$.

S'il existe un chemin joignant deux sommets a et b : $a \to b$.

Si $a \to b$ et $b \to a$: a et b sont fortement connectés : $a \leftrightarrow b$.

On impose $\leftrightarrow$ réflexif ($a \leftrightarrow a$), "être fortement connecté" est une relation d'équivalence sur V.

Forte connexité

Remarque

Une classe d'équivalence pour $\leftrightarrow$ est une composante fortement connexe (ou f. connexe) de G.
Si $V/\leftrightarrow$ contient une seule classe, on dira que G est fortement connexe (ou f. connexe).

Les sommets appartenant à un cycle maximal constituent une composante f. connexe.

Un multi-graphe orienté G est f. connexe SSI il existe un cycle passant par chaque sommet de celui-ci.

Simple connexité

Si on supprime l'orientation des arcs de G et si le multi-graphe non orienté obtenu de cette manière est connexe, alors on dira que G est simplement connexe (ou s. connexe).
$\Rightarrow$ composantes simplement connexes (ou s. connexes de G).

DÉFINITION...

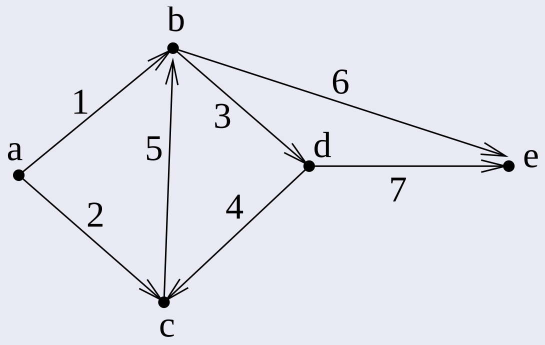
distance et diamètre définis dans le cas non orienté s'adaptent au cas d'un multi-graphe orienté fortement connexe. On remarquera cependant qu'ici, la fonction

$$d(\cdot, \cdot)$$

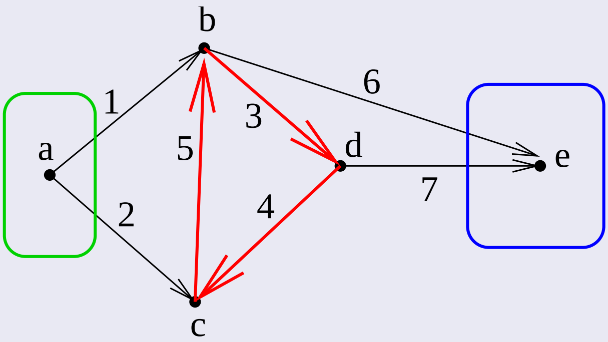
n'est en général pas symétrique.

Connexité

Exemple, s. connexe ? f. connexe ?



Exemple, s. connexe ? f. connexe ?



Recherche du plus court chemin

Soit $G = (V, E)$ un digraphe pondéré par $p : E \to \mathbb{R}^+$.
(L'algorithme s'applique aussi à un graphe non orienté.)

plus court chemin = chemin de poids minimal

se restreindre à un digraphe simple

Remarque

On suppose p à valeurs dans $\mathbb{R}^+ \cup \{+\infty\}$, on étend p de E à $V \times V$ en posant $p(x, x) = 0$, pour tout $x \in V$ et $p(x, y) = +\infty$, si $(x, y) \notin E$.

Algorithme de Dijkstra

Intuitivement, u est fixé, pour tout sommet v,

- $\mathtt{T}(v)$ initialisé à $p(u, v)$
- liste de sommets $\mathtt{C}(v)$ censée correspondre à un chemin de u à v

Lorsque l'algorithme s'achève, $\mathtt{T}(v)$ contient le poids minimal des chemins joignant u à v et $\mathtt{C}(v)$ réalise un tel chemin (ou alors, $\mathtt{T}(v) = +\infty$ si $u \not\to v$).

Idée : construire de proche en proche $X \subseteq V$ t.q. un chemin de poids minimal de u à $v \in X$ passe uniquement par des sommets de X.

X est initialisé à $\{u\}$ et à chaque étape, on ajoute un sommet à l'ensemble.

Algorithme de Dijkstra

Pour tout $v \in V$, $\mathtt{T}(v)$:=$p(u,v)$, $\mathtt{C}(v)$:=(u,v)

X :=$\{u\}$

Tant que X$\neq V$, répéter

  Choisir $v \in V \backslash$X t.q. $\forall y \in V \backslash$X, $\mathtt{T}(v) \leq \mathtt{T}(y)$

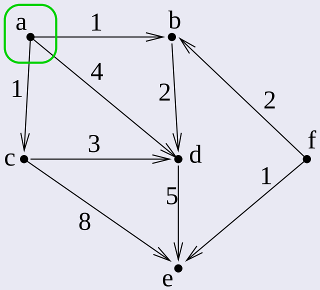
  X :=X$\cup\{v\}$

  Pour tout $y \in V \backslash$X

    Si $\mathtt{T}(y)$>$\mathtt{T}(v) + p(v,y)$,

      alors $\mathtt{T}(y)$:=$\mathtt{T}(v) + p(v,y)$ et $\mathtt{C}(y)$:=[$\mathtt{C}(v)$,y]

EXEMPLE

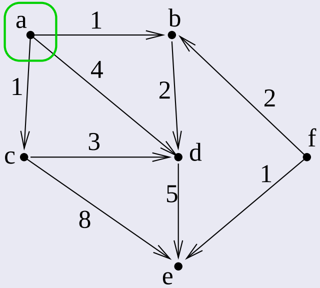


$X = \{a\}$

$$\begin{array}{c|cccccc} v & \color{blue}{a} & b & c & d & e & f \\ \hline T(v) & 0 & 1 & 1 & 4 & +\infty & +\infty \\ C(v) & (a,a) & (a,b) & (a,c) & (a,d) & (a,e) & (a,f) \end{array}.$$

Algorithme de Dijkstra

Exemple

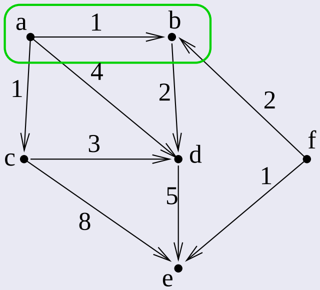


$X = \{a\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	4	$+\infty$	$+\infty$
$C(v)$	(a,a)	(a,b)	(a,c)	(a,d)	(a,e)	(a,f)

.

Exemple

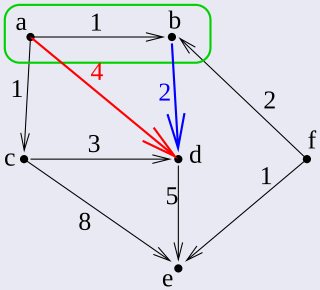


$X = \{a, b\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	4	$+\infty$	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, d)	(a, e)	(a, f)

.

EXEMPLE

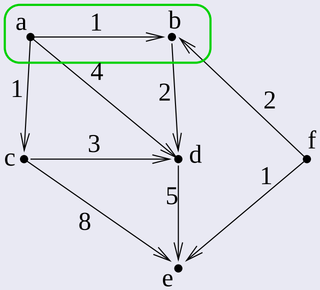


$X = \{a, b\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	4	$+\infty$	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, d)	(a, e)	(a, f)

.

EXEMPLE



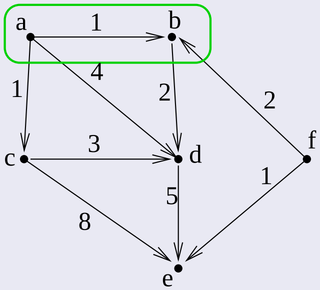
$X = \{a, b\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	$+\infty$	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, e)	(a, f)

.

Algorithme de Dijkstra

Exemple



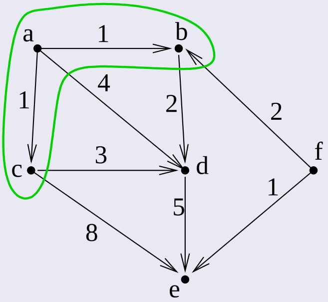
$\mathsf{X} = \{a, b\}$

v	$\color{blue}a$	$\color{blue}b$	c	d	e	f
$\mathsf{T}(v)$	0	1	$\color{red}1$	3	$+\infty$	$+\infty$
$\mathsf{C}(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, e)	(a, f)

.

Algorithme de Dijkstra

Exemple

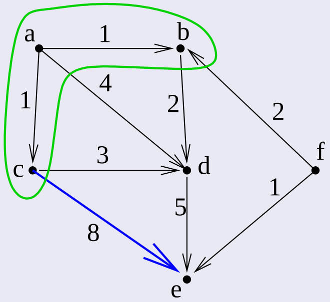


$X = \{a, b, c\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	$+\infty$	$+\infty$
$C(v)$	(a,a)	(a,b)	(a,c)	(a,b,d)	(a,e)	(a,f)

.

Exemple

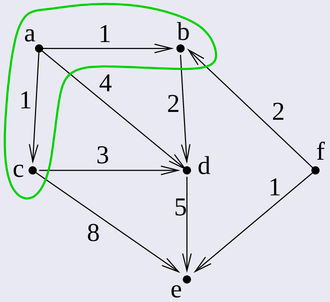


$X = \{a, b, c\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	$+\infty$	$+\infty$
$C(v)$	(a,a)	(a,b)	(a,c)	(a,b,d)	(a,e)	(a,f)

.

EXEMPLE

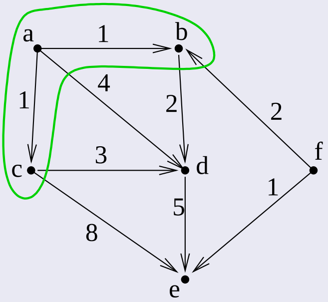


$X = \{a, b, c\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	9	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, c, e)	(a, f)

.

EXEMPLE

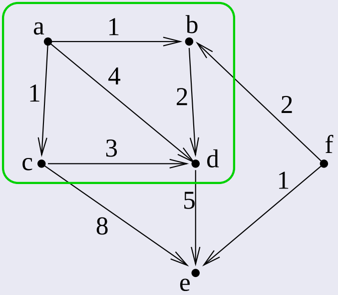


$X = \{a, b, c\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	9	$+\infty$
$C(v)$	(a,a)	(a,b)	(a,c)	(a,b,d)	(a,c,e)	(a,f)

.

EXEMPLE



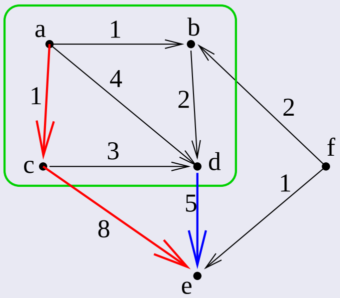
$\mathsf{X} = \{a, b, c, d\}$

v	a	b	c	d	e	f
$\mathsf{T}(v)$	0	1	1	3	9	$+\infty$
$\mathsf{C}(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, c, e)	(a, f)

.

Algorithme de Dijkstra

Exemple

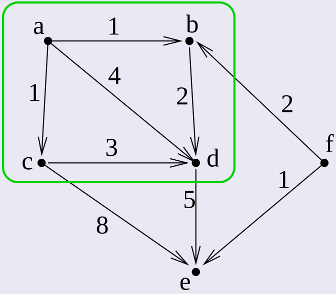


$X = \{a, b, c, d\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	9	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, c, e)	(a, f)

.

EXEMPLE



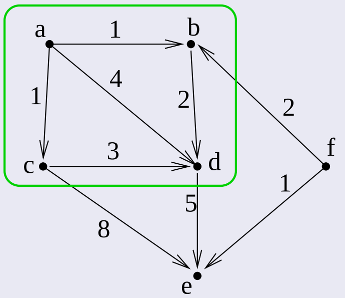
$X = \{a, b, c, d\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	8	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, b, d, e)	(a, f)

.

Algorithme de Dijkstra

Exemple

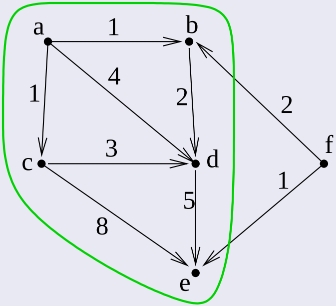


$X = \{a, b, c, d\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	8	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, b, d, e)	(a, f)

.

EXEMPLE



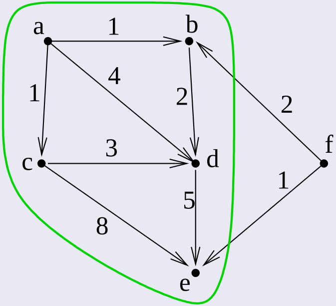
$\mathsf{X} = \{a, b, c, d, e\}$

v	a	b	c	d	e	f
$\mathsf{T}(v)$	0	1	1	3	8	$+\infty$
$\mathsf{C}(v)$	(a,a)	(a,b)	(a,c)	(a,b,d)	(a,b,d,e)	(a,f)

.

Algorithme de Dijkstra

Exemple

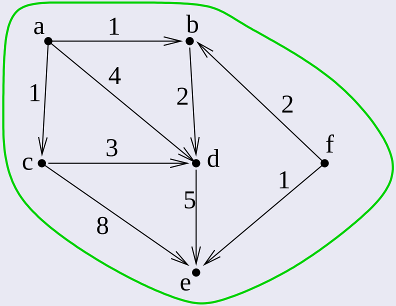


$X = \{a, b, c, d, e\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	8	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, b, d, e)	(a, f)

.

EXEMPLE



$X = \{a, b, c, d, e, f\}$

v	a	b	c	d	e	f
$T(v)$	0	1	1	3	8	$+\infty$
$C(v)$	(a, a)	(a, b)	(a, c)	(a, b, d)	(a, b, d, e)	(a, f)

.

ALGORITHME DE DIJKSTRA

```
Pour tout v ∈ V, T(v) :=p(u,v), C(v) :=(u,v)
X :={u}
Tant que X≠ V, répéter
   Choisir v ∈ V\X t.q. ∀y ∈ V\X, T(v) ≤T(y)
   X :=X∪{v}
   Pour tout y ∈ V\X
      Si T(y)>T(v) + p(v,y),
         alors T(y) :=T(v) + p(v,y) et C(y) :=[C(v),y]
```

L'algorithme s'achève : à chaque itération de la boucle, un nouvel état est ajouté à X.

Thèse : Il s'achève avec le résultat attendu.

Récurrence sur $\#X$, on vérifie I) et II) pour toutes les valeurs de $\#X$, d'où le résultat pour $X = V$.

I) $\forall v \in X$, $\mathrm{T}(v)$ est le poids minimal de tous les chemins joignant u à v.

II) $\forall v \notin X$, $\mathrm{T}(v)$ est le poids minimal des chemins joignant u à v qui, à l'exception de v, passent uniquement par des sommets de X.

Pour $\#X = 1$, c'est immédiat car $X = \{u\}$ et l'initialisation effectuée dans l'algorithme correspond aux assertions.

OK pour $\#X = n$ et ? OK ? pour $\#X = n + 1$, $1 \leq n < \#V$.

Algorithme de Dijkstra

Si $\#X = n$, Dijkstra : ajouter un sommet v ayant une valeur $\mathtt{T}(v)$ minimale parmi les sommets n'appartenant pas à X,

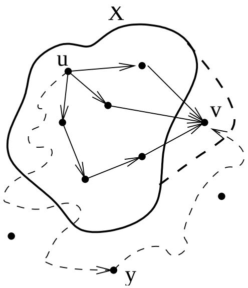
$$\mathtt{Choisir}\ v \in V\backslash\mathtt{X}\ \mathtt{t.q.}\ \forall y \in V\backslash\mathtt{X},\ \mathtt{T}(v) \leq \mathtt{T}(y)$$

Par l'hypothèse de récurrence II) :

II) $\forall v \notin X$, $\mathtt{T}(v)$ est le poids minimal des chemins joignant u à v qui, à l'exception de v, passent uniquement par des sommets de X.

A ce stade $\#X = n$ et $v \notin X$, donc
$\mathtt{T}(v)$ est le poids minimal des chemins joignant u à v qui, à l'exception de v, passent uniquement par des sommets de X.

ALGORITHME DE DIJKSTRA



Ajoutons v à X pour obtenir un ensemble de taille $n+1$.

P.A. Supposons I) non vérifié. Supposons que $\mathrm{T}(v)$ n'est pas le poids minimal des chemins joignant u à v.

Alors, $\exists y \notin X$ et un chemin de u à v p.p. y de poids $<\mathrm{T}(v)$.

Donc $\mathrm{T}(y) < \mathrm{T}(v)$ Impossible ! vu choix du sommet v.

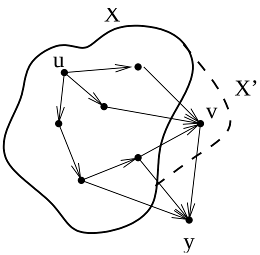
Démontrons II) de la même manière.

On passe d'un ensemble X à n sommets à un ensemble X' à $n+1$ sommets en ajoutant un sommet v.

P.A. Supposons que II) n'est plus satisfait pour X',

II) $\forall v \notin X$, $\mathrm{T}(v)$ est le poids minimal des chemins joignant u à v qui, à l'exception de v, passent uniquement par des sommets de X.

Supposons qu'il existe un sommet $y \notin X'$ tel que $\mathrm{T}(y)$ est $>$ au poids minimal des chemins joignant u à y qui, à l'exception de y, passent uniquement par des sommets de X'.



Or, par hypothèse de récurrence, II) est satisfait pour $\#X = n$.

Autrement dit, avant d'ajouter le sommet v, $\mathrm{T}(y)$ était minimal pour les chemins joignant u à y qui, à l'exception de y, passent uniquement par des sommets de X.

Ainsi, en ajoutant le sommet v à X, on aurait remplacé $\mathtt{T}(y)$ par une valeur supérieure, ce qui est en contradiction avec les prescriptions de l'algorithme :

$$\mathtt{Si\ T}(y)\mathtt{>T}(v)+p(v,y)\mathtt{,\ alors\ T}(y)\mathtt{\ :=T}(v)+p(v,y)\mathtt{\ et\ C}(y)\mathtt{\ :=[C}(v)\mathtt{,}y\mathtt{]}$$

COMPLEXITÉ

l'algorithme de Dijkstra a une complexité temporelle en $\mathcal{O}((\#V)^2)$.

Avec une implémentation minutieuse, utilisant les listes d'adjacence et les files de priorité, on obtient même une complexité en $\mathcal{O}((\#E + \#V)\log \#V)$.

APPLICATION

Routage de paquets sur Internet. Lorsque chaque paquet passe par une passerelle, un routeur étudie où le paquet doit se rendre et décide alors de la passerelle suivante où l'envoyer. Le but de chaque routeur est de propager un paquet vers un point de plus en plus près de sa destination finale.

Un des types de routage : routage SPF (Shortest Path First) chaque routeur gère sa propre carte de l'internet afin de mettre à jour sa table de routage en calculant les plus courts chemins entre lui et les autres destinations. Sa carte est un graphe orienté et pondéré, dont les sommets sont les passerelles et les arcs les connexions entre celles-ci. Chaque arc est pondéré par les dernières performances constatées sur la connexion.

graphe connexe : pathologies de graphes possédant des sommets isolés.

DÉFINITION

Un chemin (resp. un circuit) d'un multi-graphe G est eulérien s'il passe une et une seule fois par chaque arête/arc de G.

Un tel chemin peut passer plus d'une fois par un même sommet.

Un chemin (resp. un circuit) eulérien est une piste (resp. une piste fermée) passant par chaque arête/arc de G.

Un multi-graphe eulérien est un graphe qui possède un circuit eulérien.

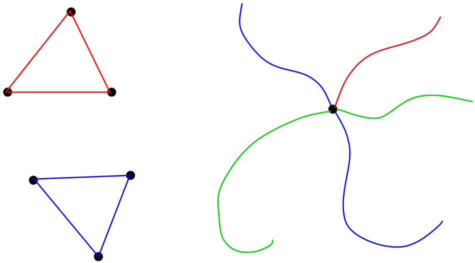
REMARQUE

Déterminer si un graphe G possède ou non un chemin eulérien, les boucles de G n'ont aucune importance.

GRAPHES EULÉRIENS

CONDITION NÉCESSAIRE

Pour que G possède un circuit eulérien, i.e., pour que G soit eulérien, il est nécessaire que G soit connexe et que le degré de chaque sommet soit pair.



Condition suffisante (propriété locale

THÉORÈME

Un multi-graphe fini non orienté connexe $G = (V, E)$ possède un circuit eulérien SSI le degré de chaque sommet est pair.

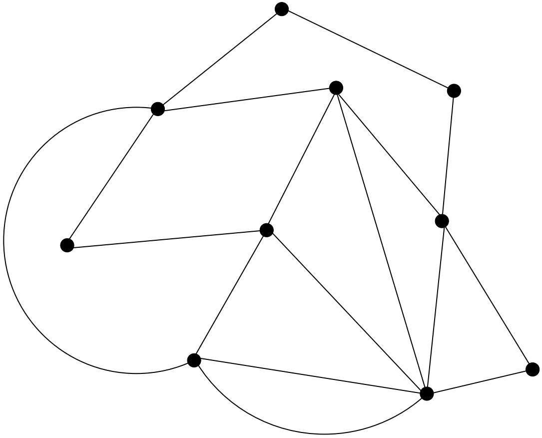
$\Rightarrow$ Hyp. : chaque sommet est de degré pair.

Construction d'une piste à partir d'un sommet a_1 de G.
A chaque étape $i \geq 1$, choix d'un sommet a_{i+1} t.q. une arête $\{a_i, a_{i+1}\} \in E$ est sélectionnée parmi les $\#E - i + 1$ arêtes non déjà sélectionnées.

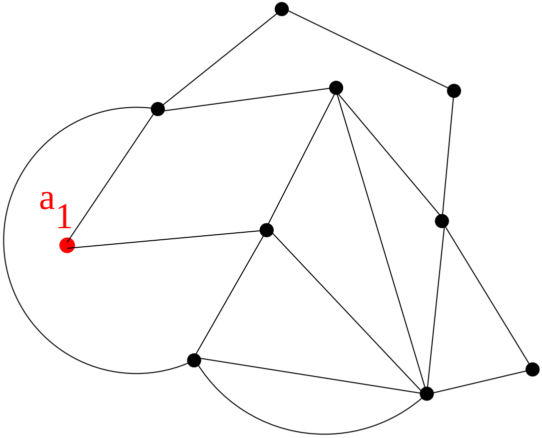
• sélection toujours possible : chaque sommet est de degré pair, "*lorsqu'on aboutit dans un sommet, on peut toujours en repartir*".

• la procédure s'achève : le graphe est fini.

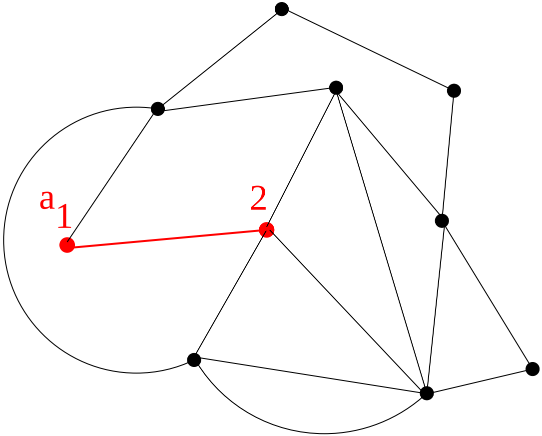
Graphes eulériens



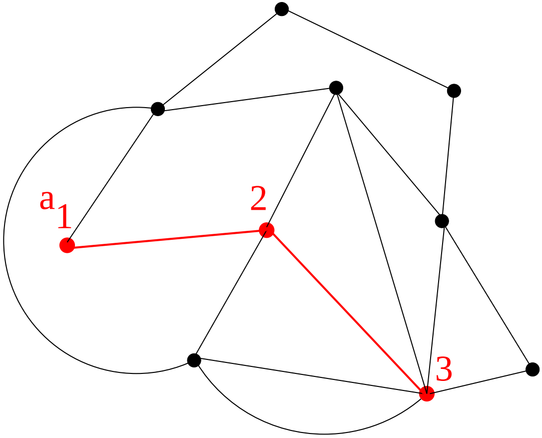
Graphes eulériens



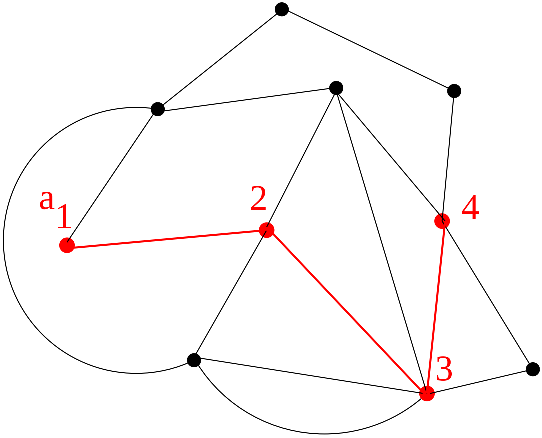
GRAPHES EULÉRIENS



Graphes eulériens



GRAPHES EULÉRIENS

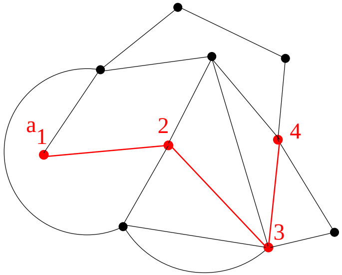


Graphes eulériens

On dispose d'une piste P joignant a_1 à un certain a_ℓ.

On peut supposer que cette piste est fermée, i.e., $a_\ell = a_1$.

Si a_ℓ diffère de a_1, puisque le degré de chaque sommet est pair, on peut étendre la piste en ajoutant une arête $\{a_\ell, a_{\ell+1}\}$. En continuant de la sorte, on épuise les sommets jusqu'à revenir en a_1.

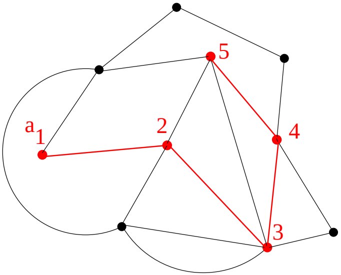


Si la piste fermée P est un circuit eulérien, le théorème est démontré.

On dispose d'une piste P joignant a_1 à un certain a_ℓ.

On peut supposer que cette piste est fermée, i.e., $a_\ell = a_1$.

Si a_ℓ diffère de a_1, puisque le degré de chaque sommet est pair, on peut étendre la piste en ajoutant une arête $\{a_\ell, a_{\ell+1}\}$. En continuant de la sorte, on épuise les sommets jusqu'à revenir en a_1.

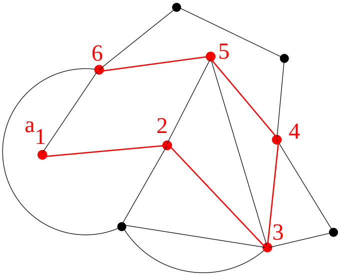


Si la piste fermée P est un circuit eulérien, le théorème est démontré.

On dispose d'une piste P joignant a_1 à un certain a_ℓ.

On peut supposer que cette piste est fermée, i.e., $a_\ell = a_1$.

Si a_ℓ diffère de a_1, puisque le degré de chaque sommet est pair, on peut étendre la piste en ajoutant une arête $\{a_\ell, a_{\ell+1}\}$. En continuant de la sorte, on épuise les sommets jusqu'à revenir en a_1.



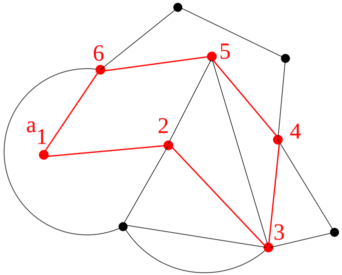
Si la piste fermée P est un circuit eulérien, le théorème est démontré.

GRAPHES EULÉRIENS

On dispose d'une piste P joignant a_1 à un certain a_ℓ.

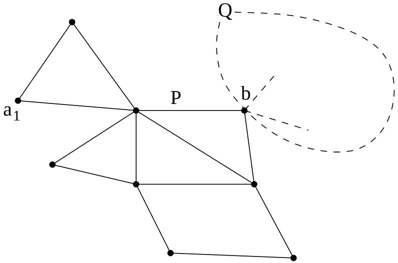
On peut supposer que cette piste est fermée, i.e., $a_\ell = a_1$.

Si a_ℓ diffère de a_1, puisque le degré de chaque sommet est pair, on peut étendre la piste en ajoutant une arête $\{a_\ell, a_{\ell+1}\}$. En continuant de la sorte, on épuise les sommets jusqu'à revenir en a_1.



Si la piste fermée P est un circuit eulérien, le théorème est démontré.

Sinon, il existe un sommet b de P qui est l'extrémité d'un nombre pair d'arêtes n'apparaissant pas dans P.

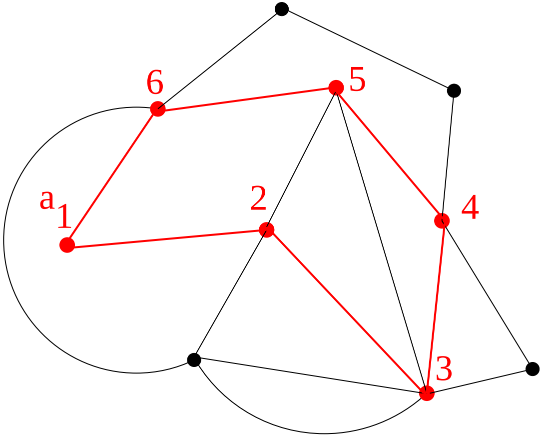


Depuis b, il est possible de construire une piste fermée Q formée uniquement d'arêtes n'apparaissant pas dans P. (même procédure, le degré de chaque sommet est encore pair.)

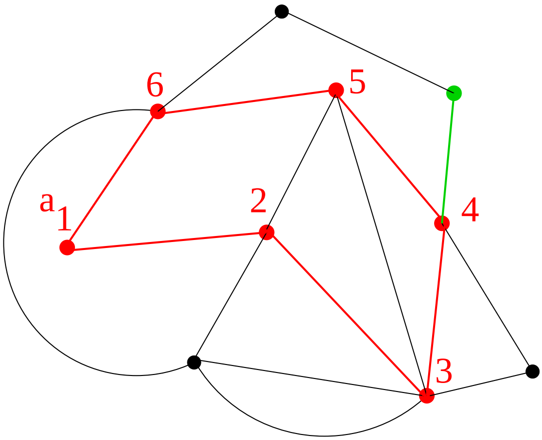
$\longrightarrow$ on étend la piste P en une piste plus longue $P \cup Q$

$\longrightarrow$ répéter cette étape un nombre fini de fois.

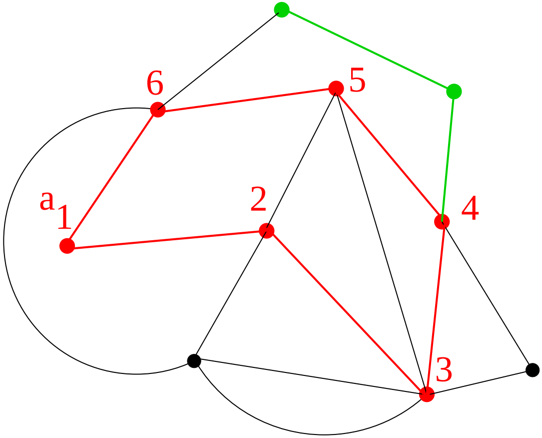
Graphes eulériens



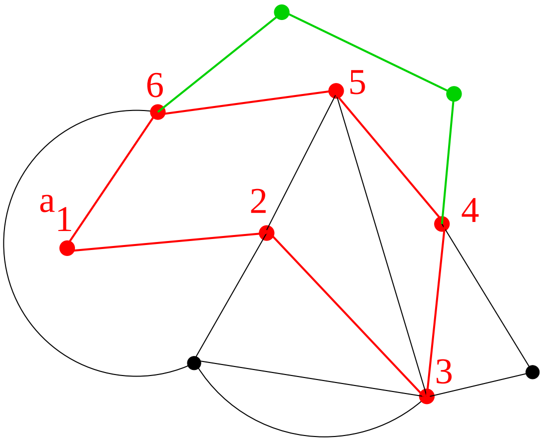
Graphes eulériens



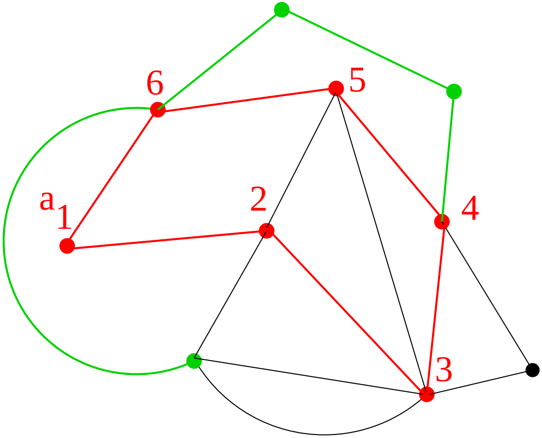
Graphes eulériens



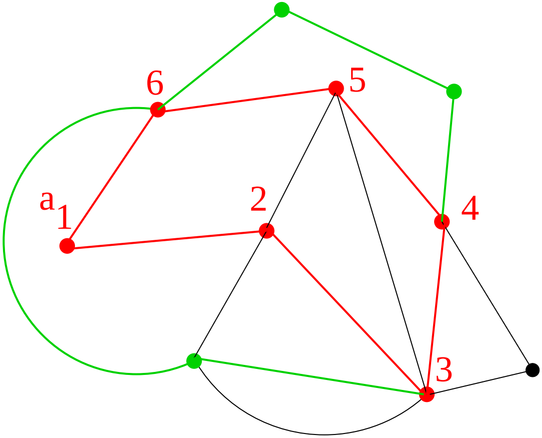
Graphes eulériens



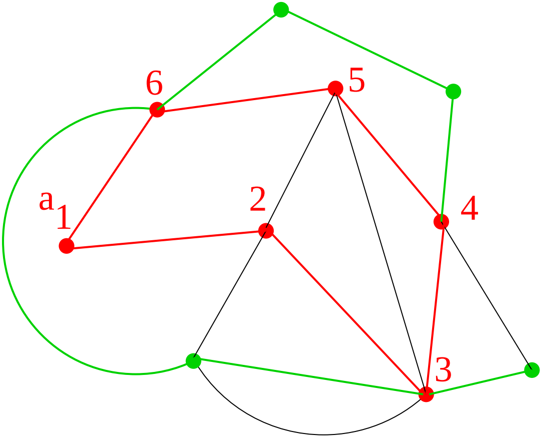
GRAPHES EULÉRIENS



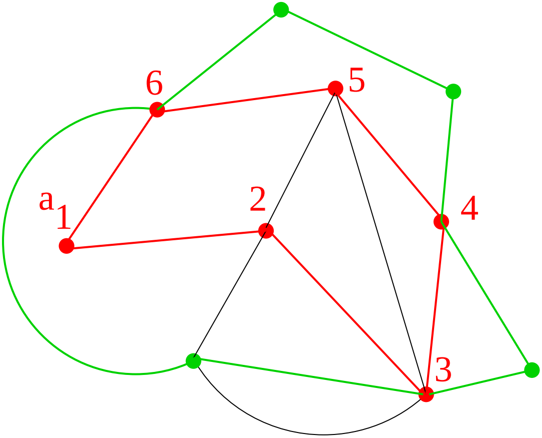
Graphes eulériens



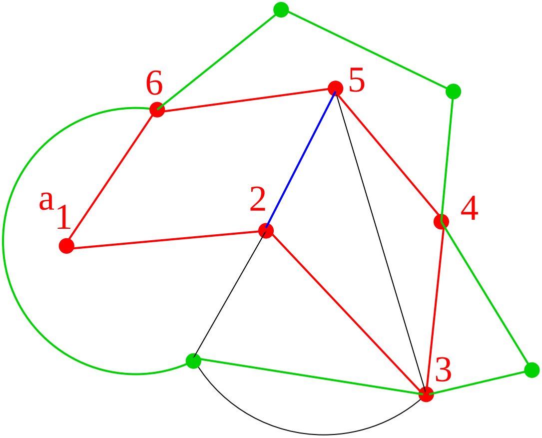
Graphes eulériens



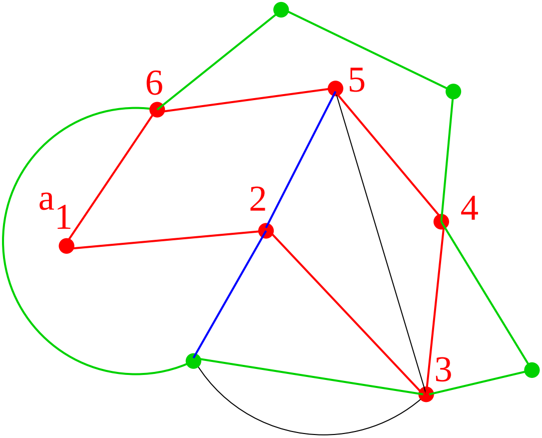
Graphes eulériens



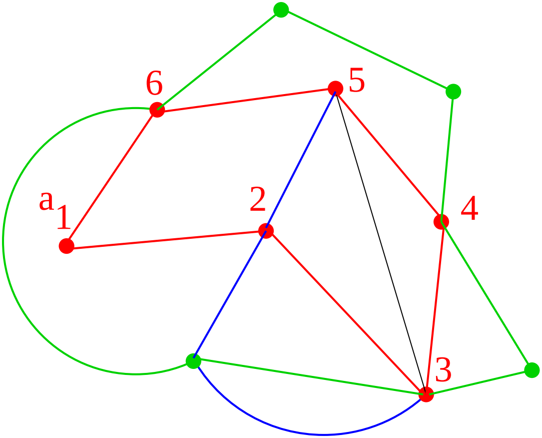
Graphes eulériens



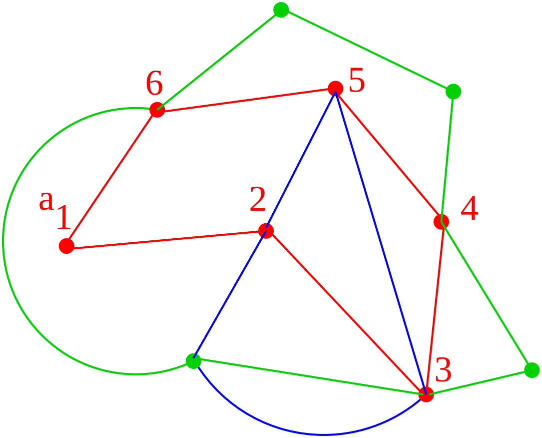
Graphes eulériens



Graphes eulériens

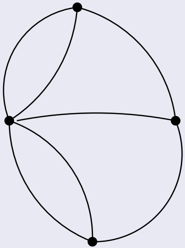


Graphes eulériens



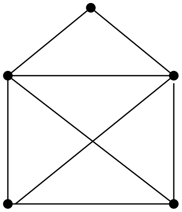
COROLLAIRE

Le problème des sept ponts de Königsberg n'admet pas de solution.



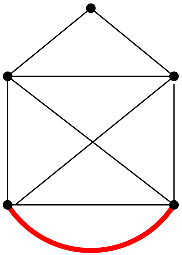
COROLLAIRE

Un multi-graphe non orienté connexe possède un chemin eulérien joignant deux sommets a et b SSI a et b sont les deux seuls sommets de degré impair.



COROLLAIRE

Un multi-graphe non orienté connexe possède un chemin eulérien joignant deux sommets *a* et *b* SSI *a* et *b* sont les deux seuls sommets de degré impair.



THÉORÈME

Un multi-graphe fini orienté s.connexe $G = (V, E)$ possède un circuit eulérien SSI $\forall v \in V$, $d^+(v) = d^-(v)$.

COROLLAIRE

Un multi-graphe fini orienté connexe $G = (V, E)$ possède un chemin eulérien SSI il existe deux sommets v_0 et v_1 tel que

- pour tout $v \in V \setminus \{v_0, v_1\}$, $d^-(v) = d^+(v)$,
- $d^+(v_0) = d^-(v_0) + 1$,
- $d^-(v_1) = d^+(v_1) + 1$.

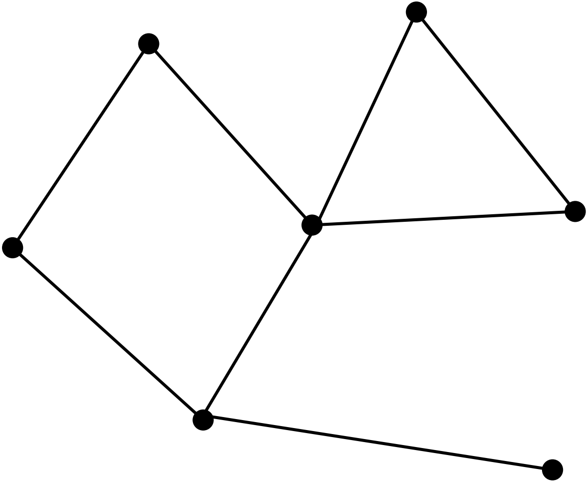
Connexité des graphes non orientés

Connexité : graphe simple suffit.

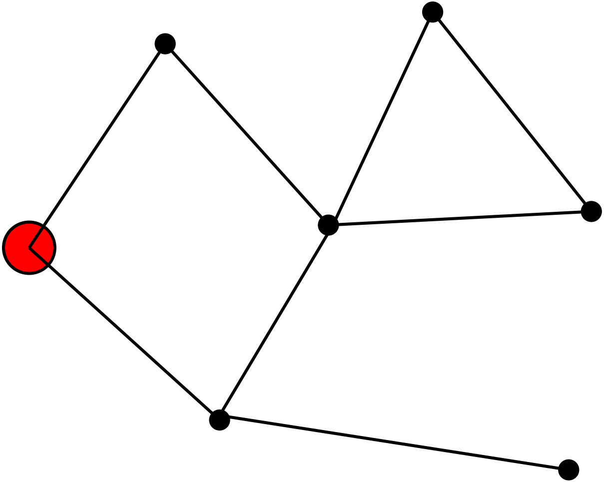
Algorithme naïf permettant de décider si un graphe non orienté est connexe.

```
Choisir au hasard un sommet v0 ∈ V
Composante := {v0},  New := {v0}
Tant que New≠ ∅
  Voisins := ∅
  pour tout sommet v appartenant à New
    Voisins := Voisins∪ν(v)
  New := Voisins\Composante
  Composante := Composante∪New
Si Composante=V
  alors sortie : "oui, G connexe"
  sinon sortie : "non, G non connexe"
```

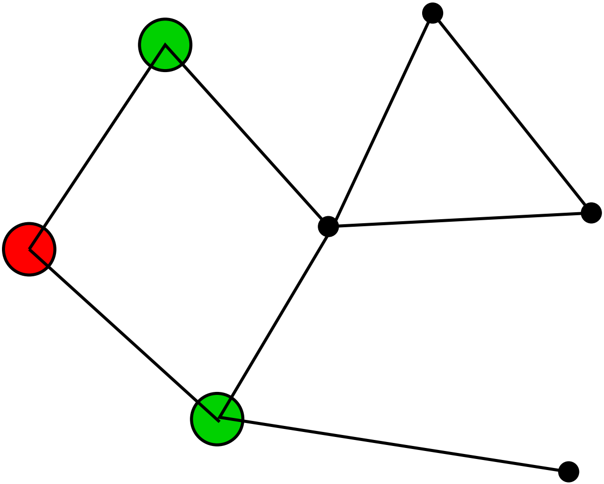
Connexité des graphes non orientés



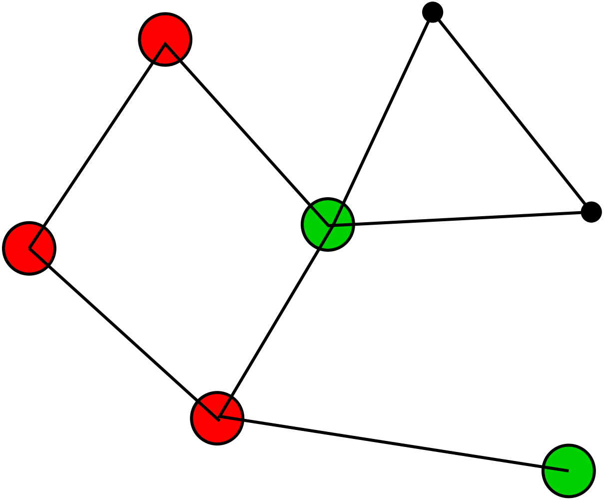
Connexité des graphes non orientés



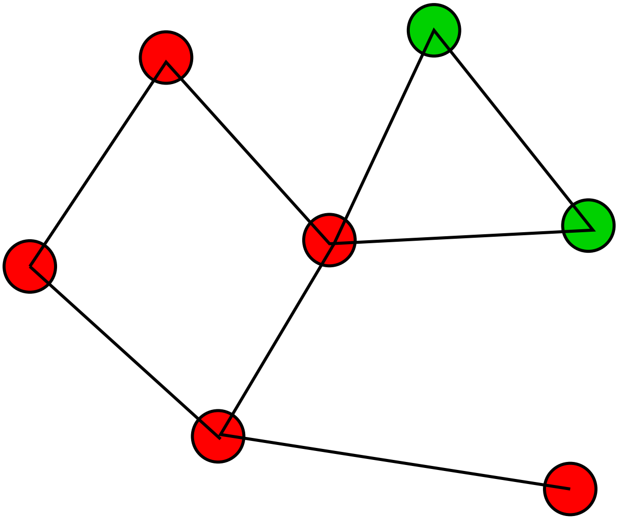
Connexité des graphes non orientés



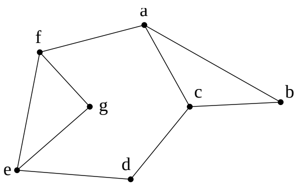
Connexité des graphes non orientés



CONNEXITÉ DES GRAPHES NON ORIENTÉS



Connexité des graphes non orientés



le dictionnaire des voisins

v	$\nu(v)$
a	$\{b, c, f\}$
b	$\{a, c\}$
c	$\{a, b, d\}$
d	$\{c, e\}$
e	$\{d, f, g\}$
f	$\{a, e, g\}$
g	$\{e, f\}$

Connexité des graphes non orientés

$v_0 = c$.

Composante	New	Voisins
$\{c\}$	$\{c\}$	$\emptyset$ $\nu(c) = \{a, b, d\}$
$\{c\} \cup \{a, b, d\}$ $= \{a, b, c, d\}$	$\{a, b, d\} \setminus \{c\}$ $= \{a, b, d\}$	$\emptyset$ $\nu(a) \cup \nu(b) \cup \nu(d)$ $= \{a, b, c, e, f\}$
$\{a, b, c, d\} \cup \{e, f\}$ $= \{a, b, c, d, e, f\}$	$\{a, b, c, e, f\} \setminus \{a, b, c, d\}$ $= \{e, f\}$	$\emptyset$ $\nu(e) \cup \nu(f)$ $= \{a, d, e, f, g\}$
$\{a, b, c, d, e, f\} \cup \{g\}$ $= \{a, b, c, d, e, f, g\}$	$\{a, d, e, f, g\} \setminus \{a, b, c, d, e, f\}$ $= \{g\}$	$\emptyset$ $\nu(g) = \{e, f\}$
$\{a, b, c, d, e, f, g\} \cup \{e, f\}$ $= \{a, b, c, d, e, f, g\}$	$\{e, f\} \setminus \{a, b, c, d, e, f, g\}$ $= \emptyset$	

CLÔTURE RÉFLEXIVE ET TRANSITIVE DE succ

$$\mathrm{succ}^*(v) := \bigcup_{j=0}^{\infty} \mathrm{succ}^j(v) \quad \text{où} \begin{cases} \mathrm{succ}^0(v) = v \\ \mathrm{succ}^{j+1}(v) = \mathrm{succ}(\mathrm{succ}^j(v)) \end{cases}.$$

$$\text{Si } G \text{ est fini,} \quad \mathrm{succ}^*(v) = \bigcup_{j=0}^{\#E} \mathrm{succ}^j(v).$$

S'il existe $k < \#E$ tel que $\mathrm{succ}^k(v) = \mathrm{succ}^{k+1}(v)$, alors

$$\mathrm{succ}^*(v) = \bigcup_{j=0}^{k} \mathrm{succ}^j(v).$$

$$\forall a, b \in V, \quad a \to b \Leftrightarrow b \in \mathrm{succ}^*(a).$$

CLÔTURE RÉFLEXIVE ET TRANSITIVE DE pred

Idem avec pred^*.

$$\forall a, b \in V, \quad b \to a \Leftrightarrow b \in \mathrm{pred}^*(a).$$

$$a \leftrightarrow b \Leftrightarrow b \in \mathrm{succ}^*(a) \cap \mathrm{pred}^*(a).$$

ALGORITHME

L'algorithme précédent peut facilement être adapté pour calculer $\mathrm{succ}^*(a)$ (resp. $\mathrm{pred}^*(a)$). Il suffit principalement d'initialiser les variables `Composante` et `New` à $\{a\}$ et de remplacer $\nu(v)$ par $\mathrm{succ}(v)$ (resp. $\mathrm{pred}(v)$).

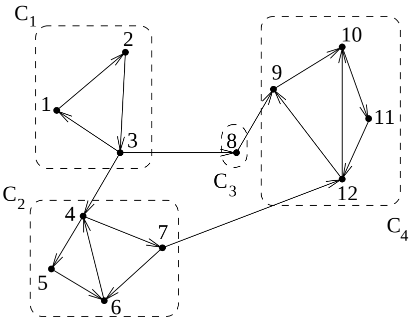
En recherchant l'intersection des deux ensembles, on détermine la composante f. connexe du sommet a.

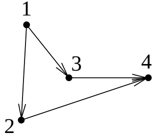
DÉCOMPOSITION EN COMPOSANTES F. CONNEXES

DÉFINITION

Soit $G = (V, E)$ un graphe orienté.

graphe acyclique des composantes ou condensé de G,
sommets = les composantes f. connexes de G.
arc entre deux composantes f. connexes A et B, s'il existe $a \in A$ et $b \in B$ tels que $a \to b$.





DÉCOMPOSITION EN COMPOSANTES F. CONNEXES

REMARQUE

S'il existe $a \in A$ et $b \in B$ tels que $a \longrightarrow b$,
alors il n'existe aucun $a' \in A$ ni aucun $b' \in B$ tels que $b' \longrightarrow a'$.

Sinon $A \cup B$ serait alors une composante f. connexe de G.
Impossible vu la maximalité des composantes connexes.

le graphe des composantes ne contient pas de cycle.

DÉCOMPOSITION EN COMPOSANTES F. CONNEXES

THÉORÈME

Soit G un digraphe (simple) t.q. $\forall v \in V, d^+(v) = d^-(v)$.
Alors, G est f. connexe SSI il est s. connexe.

f. connexe $\Rightarrow$ s. connexe.

$\Leftarrow$ G s.connexe et $\forall v \in V, d^+(v) = d^-(v)$
G possède un circuit eulérien : G est f. connexe.

SOUS-GRAPHE

DÉFINITION

Soient $G = (V, E)$ et $G' = (V', E')$ deux graphes (orientés ou non). Le graphe G' est un sous-graphe de G si

- $V' \subseteq V$,
- $E' \subseteq E \cap (V' \times V')$.

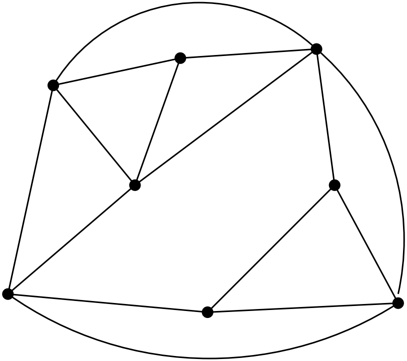
REMARQUE

Si on enlève un sommet v de G,
il faut enlever tous les arcs (ou arêtes) incidents à v.

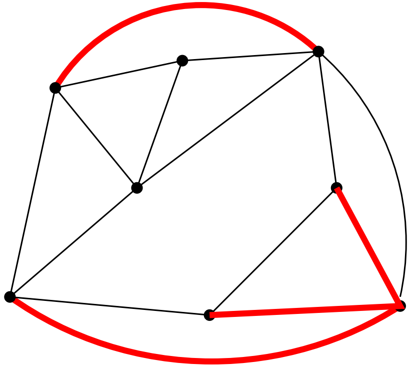
DÉFINITION

G' est un sous-graphe propre de G si $E' \subsetneq E$ ou $V' \subsetneq V$.

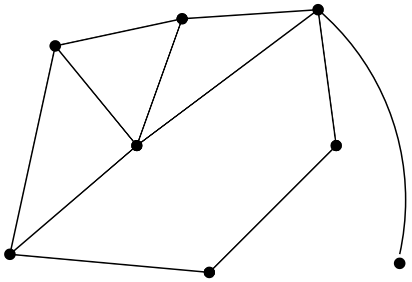
Sous-graphe



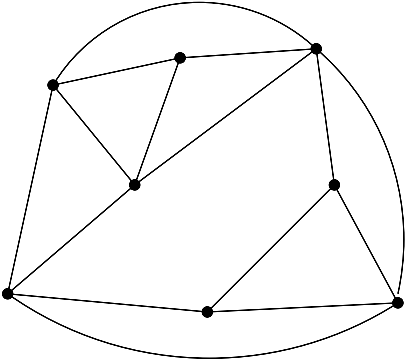
Sous-graphe



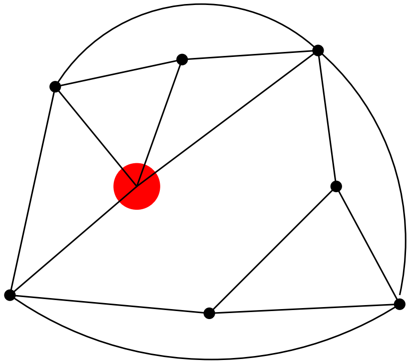
SOUS-GRAPHE



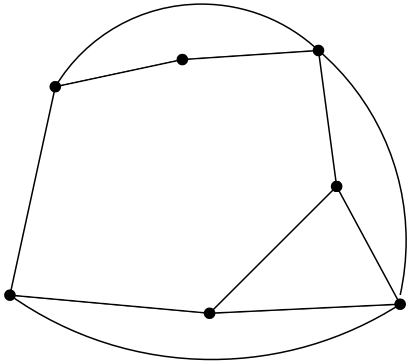
Sous-graphe



Sous-graphe



Sous-graphe



NOTATION

$G - e$ (resp. $G - v$) le sous-graphe G' de G obtenu en supprimant l'arc (ou l'arête) e (resp. le sous-graphe G' obtenu en supprimant le sommet v et les arcs (ou les arêtes) adjacents).

$G = G' + e$ (resp. $G = G' + v$) : le graphe obtenu par adjonction à G' d'une arête, d'un sommet.

Si $W = \{v_1, \ldots, v_k\} \subseteq V$, alors $G - W$ est le sous-graphe

$$(\cdots((G - v_1) - v_2) \cdots - v_{k-1}) - v_k := G - v_1 - \cdots - v_k .$$

On procède de même pour un ensemble fini d'arcs (ou d'arêtes) et on introduit une notation $G - F$ pour $F \subset E$.

Sous-graphe

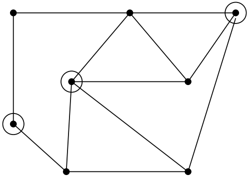
Définition

$W \subseteq V$. Le sous-graphe de G induit par W est le sous-graphe $G' = (W, E')$ avec $E' = E \cap (W \times W)$.

Définition

Si $W \subseteq V$ est tel que le sous-graphe induit par W ne contient aucune arête, alors les sommets de W sont dits indépendants.

Le nombre maximal de sommets indépendants de G : $\alpha(G)$



Sous-graphe

Définition

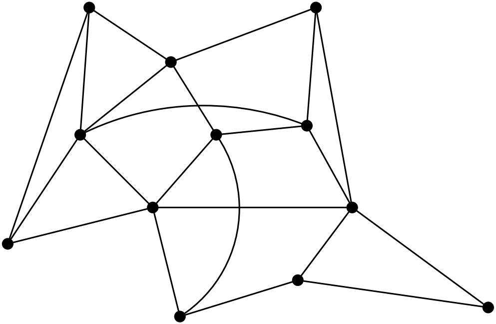
Soient $G = (V, E)$ un graphe et $G' = (V', E')$ un de ses sous-graphes.

G' est un sous-graphe couvrant G, si $V' = V$ et si

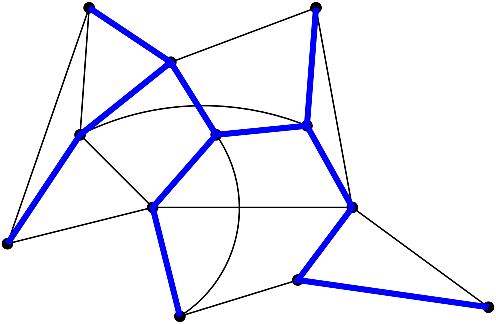
$$\forall v \in V, \exists z \in V : \{z, v\} \in E'.$$

On dira que E' est une couverture (des sommets) de G. Autrement dit, tout sommet de G est une extrémité d'une arête de E'.

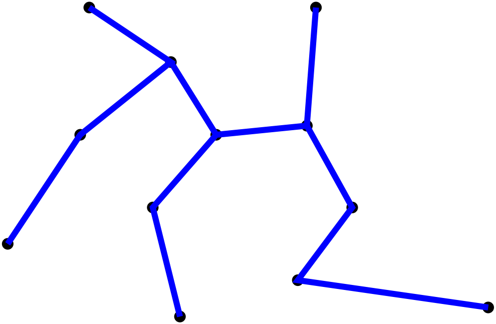
Sous-graphe



Sous-graphe



Sous-graphe



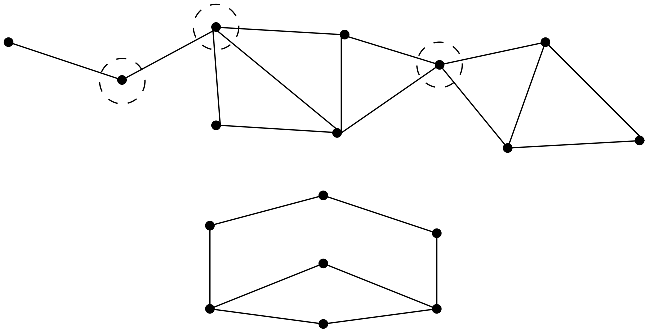
DÉFINITION

Soit $H = (V, E)$ un multi-graphe non orienté connexe (ou une composante connexe).
v est un point d'articulation, si $H - v$ n'est plus connexe (ou réduit à un point).

$\#$ comp. connexes de $H - v > \#$ comp. connexes de H.

Si H connexe et aucun point d'articulation, alors H est au moins 2-connexe.

Coupes, points d'articulation, k-connexité



Coupes, points d'articulation, k-connexité

Définition

$W \subset V$ ensemble d'articulation du multi-graphe connexe $H = (V, E)$ si $H - W$ n'est plus connexe (ou réduit à un sommet).

Pour un multi-graphe H quelconque, W est un ensemble d'articulation si $H - W$ a plus de composantes connexes que H.

Définition

Pour un graphe connexe H,
$\kappa(H)$= taille minimale d'un ensemble d'articulation de H,

$$\kappa(H) = \min\{\#W \mid W \subseteq V : H - W \text{ disconnecté}$$

$$\text{ou réduit à un sommet}\}.$$

$\kappa(K_n) = n - 1$. Si G non connexe, on pose $\kappa(G) = 0$.

Terminologie

Si $\kappa(H) = k \geq 1$, H est k-connexe.

$\kappa(G) = k$: quels que soient les $k - 1$ sommets supprimés, G reste connexe mais
il est possible de supprimer k sommets “bien choisis” pour disconnecter G (ou le rendre trivial).

graphe “au moins 2-connexe” : $\kappa(G) \geq 2$

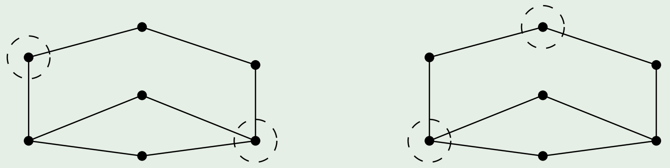
G est au moins k-connexe : $\kappa(G) \geq k$.

Remarque

$\kappa(G) = k + 1 \Rightarrow \kappa(G) \neq k$ mais $\kappa(G) \geq k + 1 \Rightarrow \kappa(G) \geq k$

Si G est au moins $(k + 1)$-connexe, alors il est au moins k-connexe

UN GRAPHE 2-CONNEXE

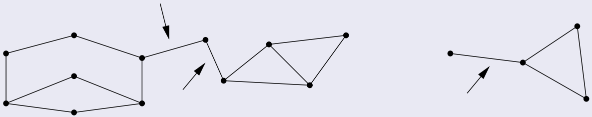


DÉFINITION

Soit $H = (V, E)$ un multi-graphe non orienté connexe (ou une composante connexe d'un multi-graphe non orienté).

e est une arête de coupure si $H - e$ n'est plus connexe.

Au moins une extrémité d'une arête de coupure est évidemment un point d'articulation de H.



Algorithme de Fleury permettant de construire un circuit eulérien (à supposer qu'un tel circuit existe).

Choisir un sommet $v_0 \in V$

i :=1

Répéter tant que possible

  Choisir une arête $e_i = \{v_{i-1}, v_i\} \in V$ telle que

- $e_i \neq$ arêtes déjà choisies $e_1, \ldots, e_{i-1}$
- autant que possible, e_i ne doit pas être

une arête de coupure de $G_i = G - \{e_1, \ldots, e_{i-1}\}$

  i :=$i + 1$

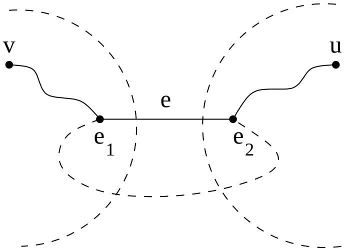
Cet algorithme fournit une suite d'arêtes $e_1, e_2, \ldots$ qui constituent un circuit eulérien.

PROPOSITION

Une arête e est une arête de coupure du graphe $H = (V, E)$ SSI e n'appartient à aucune piste fermée de H.

$\Rightarrow$ Si e est une arête de coupure
$\exists$ sommets u et v connectés dans H mais qui ne sont plus connectés dans $H - e$.

Il existe un chemin appelons-lajoignant u et v qui passe par e.



Dans $H - e$,
une partie de ce chemin joint u à une extremité de e : e_2
l'autre partie du chemin joint v à l'autre extrémité de e : e_1.

P.A. Si e appartient à une piste fermée, il existe un chemin joignant e_1 à e_2 et ne passant pas par e.
u et v sont encore connectés dans $H - e$, impossible !

Réciproque. Supposons que $e = \{e_1, e_2\}$ n'est pas une arête de coupure.

Si H est connexe, $H - e$ l'est encore.

Ainsi, il existe dans $H - e$ une piste joignant e_1 et e_2.

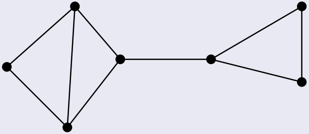
Dans H, e appartient à une piste fermée.

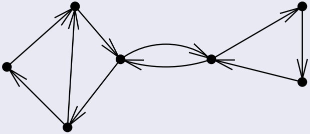
PROBLÈME DES SENS UNIQUES

Graphe non orienté connexe. On désire orienter ses arêtes pour obtenir un graphe f. connexe.

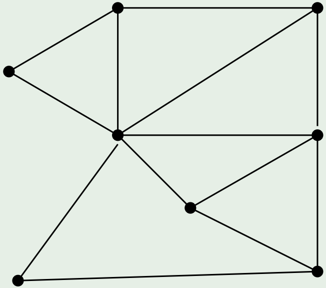
Les arêtes de coupure doivent nécessairement être remplacées par deux arcs (pas de sens unique).

Les autres arêtes appartiennent toutes à une piste fermée qu'il est aisé d'orienter (création de sens uniques).

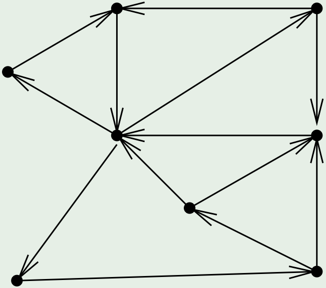




UN AUTRE EXEMPLE



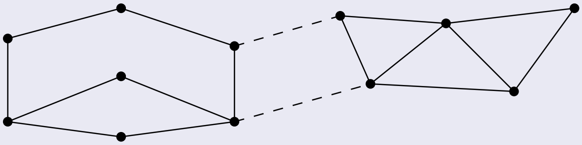
UN AUTRE EXEMPLE



DÉFINITION

Soit $H = (V, E)$ un multi-graphe non orienté connexe (ou une composante connexe d'un multi-graphe non orienté).

$F \subset E$ est un ensemble de coupure, une coupe, une coupure si F est un ensemble **minimal** (pour l'inclusion) tel que $H - F$ n'est pas connexe.



DÉFINITION

La taille minimale d'une coupe de H : $\lambda(H)$

$$\lambda(H) = \min\{\#F \mid F \subseteq E : H - F \text{ disconnecté}\}.$$

Si H n'est pas connexe : $\lambda(H) = 0$.
on rencontre souvent la notation $\kappa'(H)$.

H connexe et $\lambda(H) = k$: H est k-connexe (pour les arêtes)

Si on enlève k-1 arêtes à un graphe k-connexe (pour les arêtes), il reste connexe ; par contre, il est possible d'enlever k arêtes pour le disconnecter.

!! k-connexité $\neq$ k-connexité pour les arêtes !!

G est au moins k-connexe (pour les arêtes), si $\lambda(G) \geq k$.

Théorème de Robbins

On peut orienter un graphe connexe pour le rendre f. connexe SSI ce graphe est au moins 2-connexe pour les arêtes.

Rappels :

Problème des sens uniques

Les arêtes de coupure doivent nécessairement être remplacées par deux arcs (pas de sens unique).

Les autres arêtes appartiennent toutes à une piste fermée qu'il est aisé d'orienter (création de sens uniques).

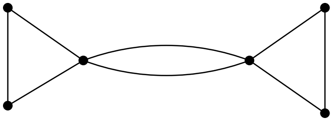
Proposition

Une arête e est une arête de coupure du graphe $H = (V, E)$ SSI e n'appartient à aucune piste fermée de H.

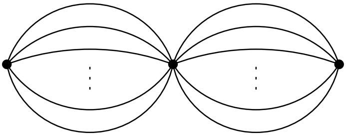
Conclusion : toute arête appartient à une piste fermée : au moins 2-connexe pour les arêtes

COUPES, POINTS D'ARTICULATION, k-CONNEXITÉ

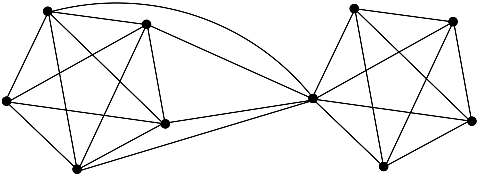
$\kappa(G) = 1$ et $\lambda(G) = 2$:



$\kappa(G) = 1$ et $\lambda(G) = k$:



Même avec un graphe simple : pas de lien entre $\lambda(G)$ et $\kappa(G)$.
Ici, $\lambda(G) = 4$ et $\kappa(G) = 1$:



REMARQUE

Si $\deg v = k$, supprimer les k arêtes incidentes à v isole v

$$\lambda(G) \leq \min_{v \in V} \deg(v).$$

THÉORÈME DE WHITNEY (1932)

$$\kappa(G) \leq \lambda(G).$$

plutôt que de supprimer une arête, il suffit d'en supprimer au plus une extrémité.

REMARQUE

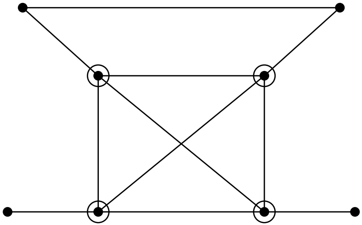
Si F est une coupure d'un graphe G connexe, alors, de par la minimalité de F, $G - F$ possède 2 composantes connexes et l'ensemble des sommets de G est partitionné en deux sous-ensembles correspondant à ces deux composantes.

Définition

Une clique d'un graphe non orienté et simple $G = (V, E)$ est un sous-graphe complet de G.

La taille d'une clique est le nombre de sommets qui la composent.

$\omega(G)$ = taille maximale d'une clique de G.



graphes simples non orientés : aucune différence pour des multi-graphes.

Définition

Soient un graphe $G = (V, E)$ et u, v deux sommets distincts de G. Un sous-ensemble $S \subseteq V \setminus \{u, v\}$ sépare u et v s'il n'existe aucun chemin joignant u et v dans le sous-graphe de G induit par $V \setminus S$.

Définition

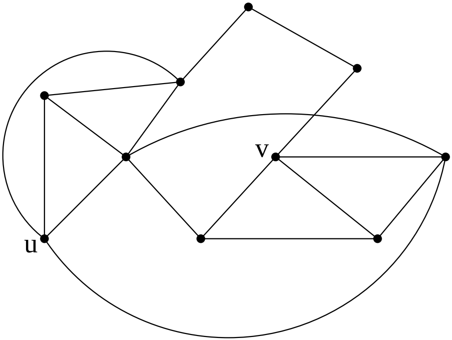
Deux chemins joignant u et v sont indépendants si les seuls sommets qu'ils ont en commun sont u et v.

Théorème(s) de Menger

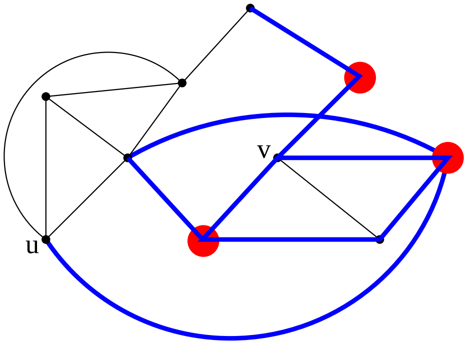
Théorème de Menger (1927)

Soient u, v deux sommets non adjacents d'un graphe connexe $G = (V, E)$. La taille minimum d'un sous-ensemble de sommets séparant u et v est égale au nombre maximum de chemins deux à deux indépendants joignant u et v.

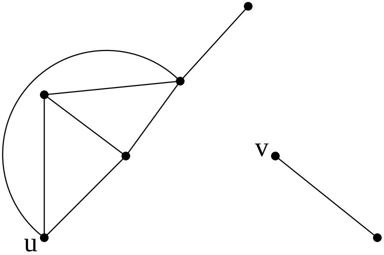
Théorème(s) de Menger



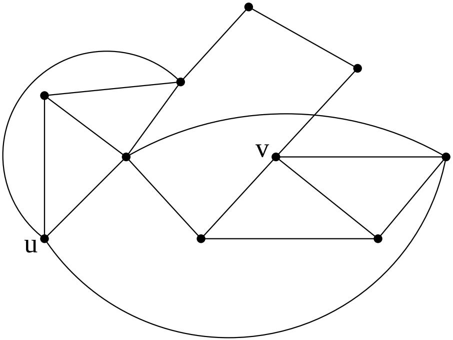
Théorème(s) de Menger



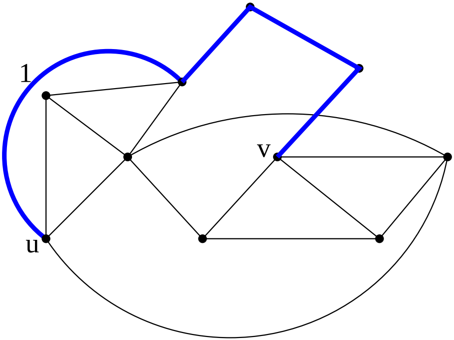
Théorème(s) de Menger



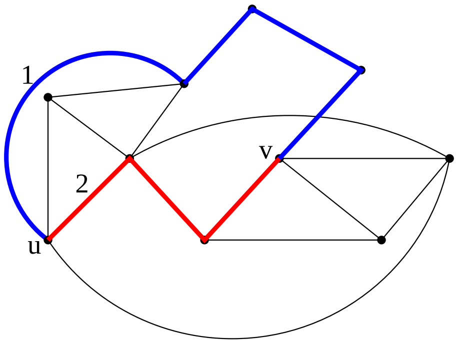
Théorème(s) de Menger



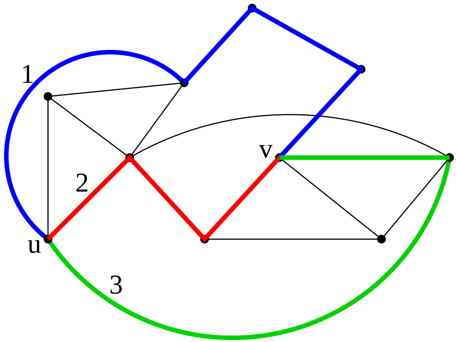
Théorème(s) de Menger



Théorème(s) de Menger



Théorème(s) de Menger



THÉORÈME(S) DE MENGER

COROLLAIRE, MENGER (1927)

Soit $k \geq 2$. Un graphe $G = (V, E)$ est au moins k-connexe (pour les sommets) SSI toute paire de sommets distincts de G est connectée par au moins k chemins indépendants.

$\Leftarrow$ Si toute paire de sommets est connectée par au moins k chemins indépendants, alors $\kappa(G) \geq k$.

THÉORÈME(S) DE MENGER

$\Rightarrow$ P.A. Supposons $\kappa(G) \geq k$ et qu'il existe 2 sommets u et v joints par au plus $k-1$ chemins indépendants.

Par le thm de Menger : u et v sont adjacents, $e = \{u, v\} \in E$.

Dans $G - e$, u et v sont joints par au plus $k-2$ chemins indépendants.

Dans $G - e$, u et v ne sont pas adjacents.
Par le thm de Menger : ils peuvent être séparés, dans $G - e$, par un ensemble S de taille minimale t.q. $\#S \leq k-2$.

Puisque $\kappa(G) \geq k$, $\#V > k$.

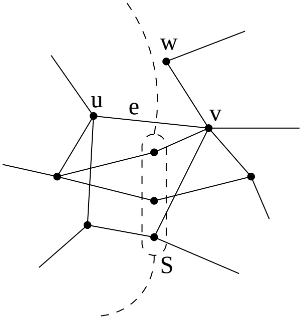
Il existe un sommet $w \notin S \cup \{u, v\}$.

RAPPEL : THÉORÈME DE MENGER (1927)

Soient u, v deux sommets non adjacents d'un graphe connexe $G = (V, E)$. La taille minimum d'un sous-ensemble de sommets séparant u et v est égale au nombre maximum de chemins deux à deux indépendants joignant u et v.

Théorème(s) de Menger

Dans $(G - e) - S$, il ne peut y avoir simultanément 2 chemins joignant w respectivement à u et à v
car sinon dans $(G - e) - S$ on aurait des chemins joignant u à w et w à v, alors u et v ne seraient pas séparés par S !
Supposons qu'aucun chemin ne joint w et u dans $(G - e) - S$.
L'ensemble $S \cup \{v\}$ possède (au plus) $k - 1$ éléments et sépare w et u dans G. Ceci contredit le fait que $\kappa(G) \geq k$.



Théorème(s) de Menger

Résultats en termes d'arêtes

Un graphe est au moins k-connexe pour les arêtes SSI toute paire de sommets est connectée par au moins k chemins ne partageant aucune arête.