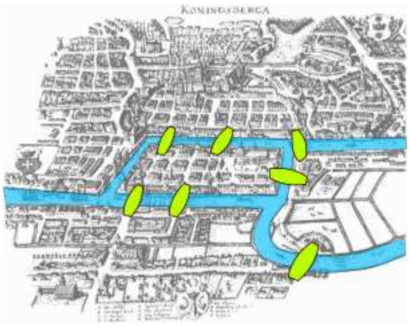
1 Introduction

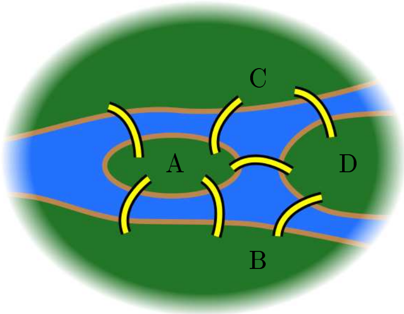
Problèmes historiquement célèbres

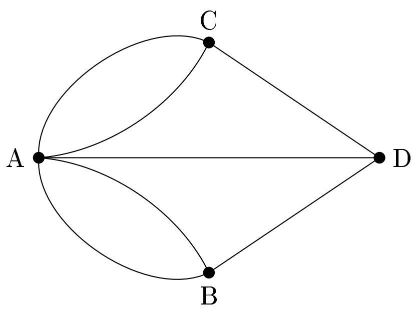
Les ponts de Könisberg

Le problème des ponts de Könisberg est à l'origine de la théorie des graphes. Cette ville de Prusse orientale est bâtie sur les bords d'une rivière ainsi que sur deux îles. Le problème consiste à trouver une promenade à partir d'un point donné qui fasse revenir à ce point en passant une fois et une seule par chacun des sept ponts de la ville de Königsberg.



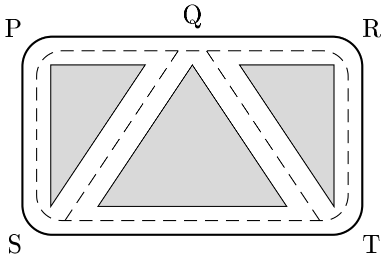
Personne n'ayant trouvé de solution, on se tourna vers le plus grand mathématicien de l'époque, Leonhard Euler. L'article qu'il publia en 1741 montre que l'on peut schématiser la situation (les deux îles, les deux bords et les sept ponts) par un diagramme que l'on appelle **graphe** :





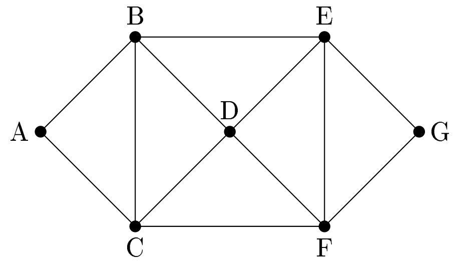
Euler a démontré que pour résoudre ce problème, il suffit de mettre en évidence certaines propriétés du graphe.

1.1 Dessiner le graphe représentant les chemins de la figure suivante et préciser le nombre de sommets, le nombre d'arêtes et le degré de chaque sommet.



Le voyageur de commerce

De prime abord, le problème du voyageur de commerce s'apparente à celui des ponts de Könisberg. Considérons par exemple 7 villes A, B, C, D, E, F et G reliées entre elles par des routes selon le diagramme ci-dessous :

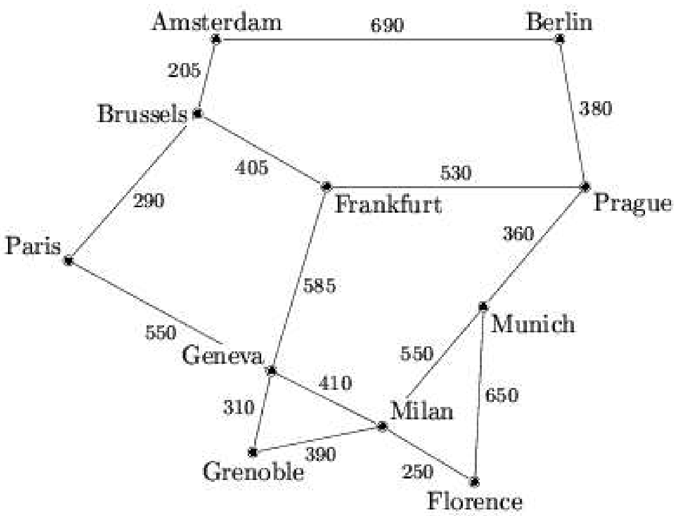


Est-il possible de trouver un trajet qui, commençant et se terminant par A, passe exactement une fois par chacune des villes B, C, D, E, F et G ?

La forme plus moderne de ce problème est attribuée à Hassler Whitney et se rapproche de la situation réelle du voyageur de commerce. Dans un graphe comme ci-dessus, on attribue à chaque arête un nombre positif (correspondant à la distance, au coût, au temps, etc.) et l'on cherche un circuit fermé minimal ne passant qu'une fois par chaque sommet.

Ce problème n'a toujours pas de solution générale, bien que des solutions dans des cas particuliers aient été trouvées.

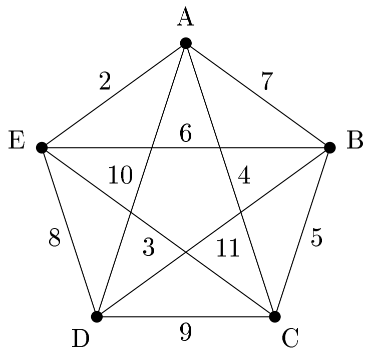
La question d'optimiser (rendre minimal ou maximal) une quantité apparaît souvent en recherche opérationnelle. Considérons par exemple un voyageur désirant aller en voiture d'Amsterdam à Florence. Il a en main le graphe suivant qui donne les distances pour relier deux villes voisines.



Quel itinéraire doit-il prendre pour rendre le trajet le plus court possible ?

La découverte d'un algorithme permettant de traiter ce genre de problèmes n'a été faite qu'en 1959.

1.2 Un gardien de zoo, commençant sa visite par les antilopes, veut encore passer chez les boas, les cerfs, les dromadaires et les éléphants. Ces lieux sont notés respectivement par A, B, C, D et E. Les distances sont indiquées sur le diagramme.



Trouver l'itinéraire le plus court possible commençant et finissant par A.

Coloriage des cartes

Considérons, par exemple, la carte des États-Unis sans l'Alaska ni Hawaï.

On colorie ce genre de cartes en sorte que deux régions (ici des États) voisines aient une couleur différente. La question de savoir combien de couleurs sont nécessaires a été posée pour la première fois par Francis Guthrie en 1852 et a fait depuis l'objet de nombreuses recherches par des mathématiciens de renom (Morgan, Cayley, Kempe).

En fait, il est assez facile de montrer que cinq couleurs suffisent toujours quelle que soit la carte, mais pas trois couleurs. On le voit en considérant l'anneau des cinq États qui entourent le Nevada. Cet anneau nécessite au moins trois couleurs et le Nevada en a besoin d'une autre, d'où un minimum de quatre.

Ainsi le problème du coloriage des cartes est devenu pendant longtemps le célèbre problème des quatre couleurs : toute carte peut-elle être coloriée avec quatre couleurs ?

Il aura fallu attendre 1976 pour que deux mathématiciens, Kenneth Appel et Wolfgang Haken, utilisant plusieurs milliers d'heures d'ordinateur, trouvent une preuve longue de plusieurs centaines de pages avec plus de 20 000 configurations. Le célèbre problème était enfin résolu.

En dualisant le problème, c'est-à-dire en remplaçant chaque pays par un sommet et chaque frontière commune par une arête entre les deux pays concernés, on obtient un graphe. Le problème du coloriage des cartes se transforme en un problème du coloriage des sommets.

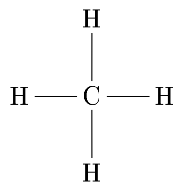
Chimie moléculaire

À l'époque de la Révolution française, on dessinait déjà des diagrammes ressemblant à des graphes pour décrire les molécules.

Mais ce n'est que dans les années 1850, sous l'influence de savants comme Auguste Kekule et Crum Brown, quand on eut compris un peu mieux la théorie des liaisons chimiques (théorie de la valence), que les graphes furent utilisés plus largement.

Cependant, de telles représentations ignorent en général la structure spatiale de la molécule.

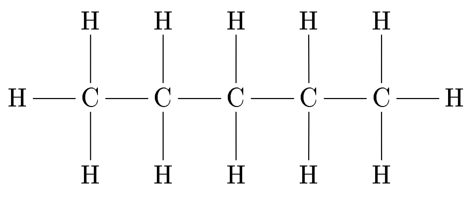
Par exemple, la molécule de méthane CH_4, représentée par le diagramme ci-contre, n'a pas une structure planaire. En effet, la stéréochimie nous enseigne que l'atome de carbone C est situé au centre d'un tétraèdre régulier dont les quatre sommets sont les atomes d'hydrogène H.



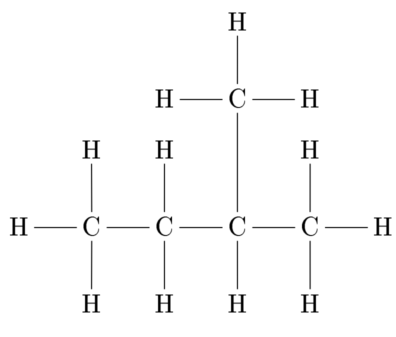
Néanmoins ce mode de représentation s'est avéré utile pour illustrer comment les différents atomes sont reliés entre eux et pour obtenir ainsi des informations sur le comportement chimique de telle ou telle molécule.

La connaissance de la formule chimique d'une molécule ne permet pas toujours de déterminer complètement toutes ses propriétés chimiques. Cela est dû à l'existence d'isomères, c'est-à-dire à des molécules ayant les mêmes atomes et les mêmes liaisons, mais arrangés différemment.

Voici, par exemple, deux isomères de C_5H_{12} :



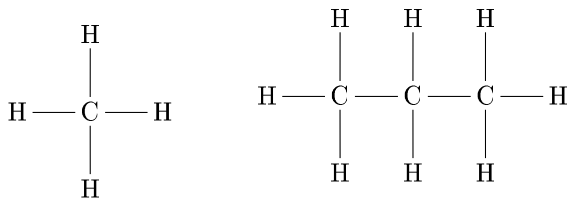
pentane



isopentane

Rappelons qu'un atome d'hydrogène H n'a qu'un électron et qu'il ne peut se lier dans une molécule qu'à un seul atome. En revanche, l'atome de carbone C a quatre électrons disponibles dans sa couche périphérique et peut se lier avec au plus quatre autres atomes.

1.3 La figure ci-dessous représente une molécule de méthane CH_4 ainsi qu'une molécule de propane C_3H_8.



méthane propane

1) En considérant ces diagrammes comme des graphes, que peut-on dire des sommets correspondant aux atomes de carbone et aux atomes d'hydrogène ?
2) Représenter les graphes correspondant aux deux isomères de C_4H_{10}.
3) Montrer que C_5H_{12} a un troisième isomère.

Autres exemples d'application

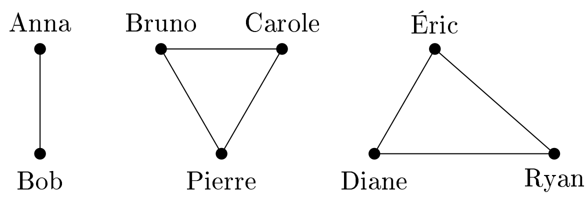
L'intérêt de la théorie des graphes réside dans sa capacité à pouvoir modéliser par des diagrammes géométriques qu'on appelle **graphes**, des relations et arrangements entre divers éléments d'une situation donnée.

L'étude des propriétés de ces diverses configurations permet de répondre à certaines questions se rapportant à la situation initiale.

Exemple

Supposons qu'à la rentrée vous vous joignez à une nouvelle classe. Vous observez que certains élèves se connaissent et d'autres pas. Une manière synthétique de présenter cette situation est de la modéliser ainsi :

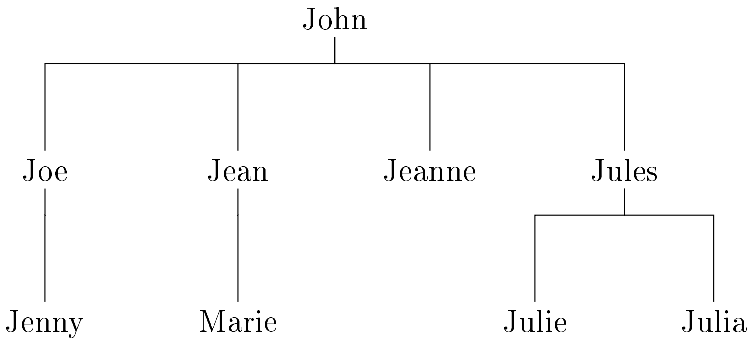
- chaque élève est représenté par un point ;
- deux élèves qui se connaissent sont reliés par un segment.



1.4 John est ami avec Johanna, Jean et Jeanne ; Joe l'est avec Jeanne et Johanna ; Jean et Johanna s'aiment l'un l'autre.

Dessiner un digraphe (c'est-à-dire un graphe dont les arêtes sont orientées) illustrant les relations entre John, Johanna, Jean, Jeanne et Joe.

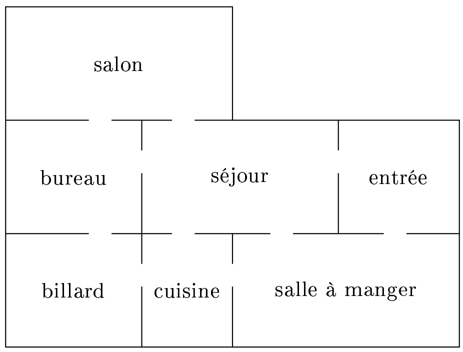
1.5 Dessiner le graphe correspondant à l'arbre généalogique suivant :



1.6 Les serpents mangent les grenouilles et les oiseaux mangent des araignées ; les oiseaux et les araignées mangent tous deux des insectes. Les grenouilles mangent des escargots, des araignées et des insectes.

Dessiner un digraphe représentant le comportement alimentaire de ces prédateurs.

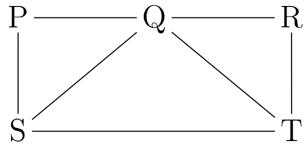
1.7 Voici le plan d'un logement :



Schématiser par un graphe le plan de circulation de ce logement.

Réponses

1.1



5 sommets 7 arêtes

sommet	P	Q	R	S	T
degré	2	4	2	3	3

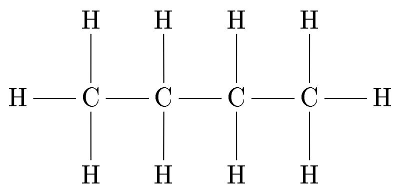
1.2 A → C → B → D → E → A ou A → E → D → B → C → A

Le second itinéraire correspond au premier itinéraire en sens inverse.

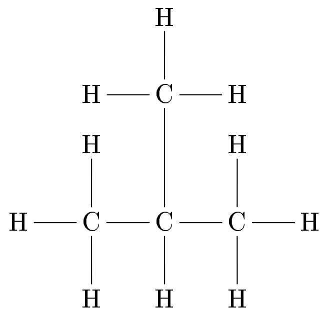
1.3

1) Les sommets correspondant aux atomes de carbone sont de degré 4 et ceux correspondant aux atomes d'hydrogène de degré 1.

2)

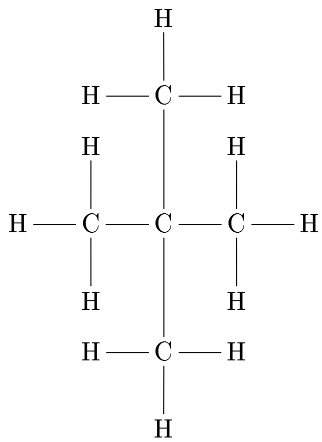


butane



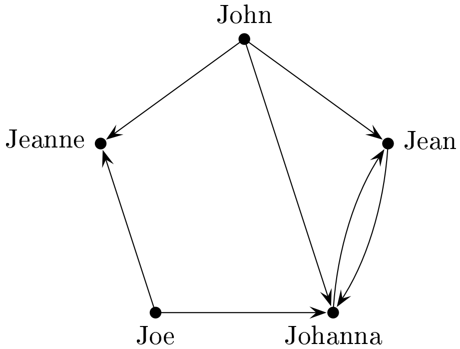
isobutane

3)

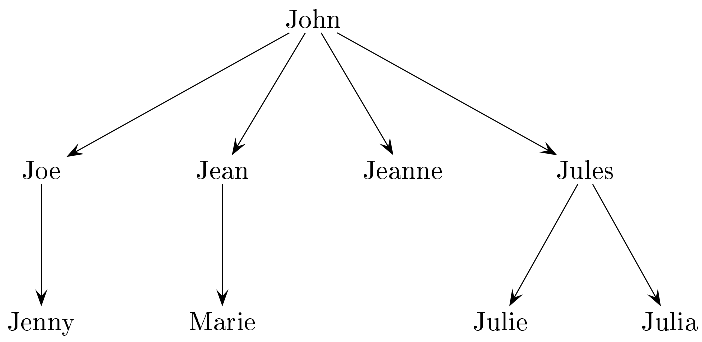


neopentane

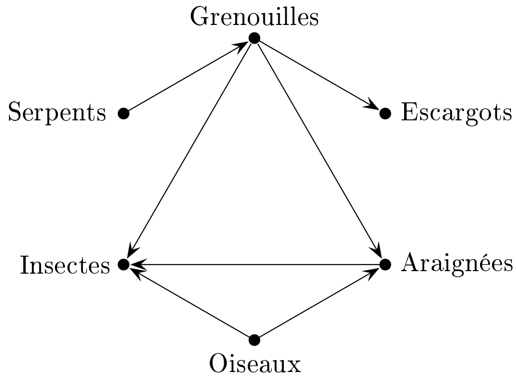
1.4



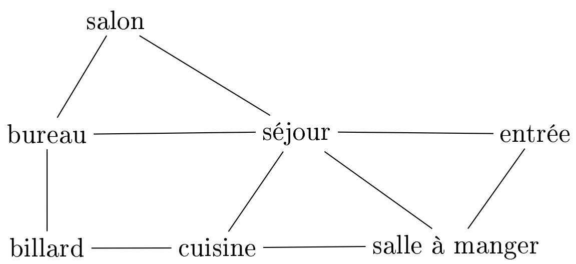
1.5


John
Joe
Jean
Jeanne
Jules
Jenny
Marie
Julie
Julia

1.6


Grenouilles
Serpents
Escargots
Insectes
Araignées
Oiseaux

1.7


salon
bureau
séjour
entrée
billard
cuisine
salle à manger

2 Concepts élémentaires

Définitions

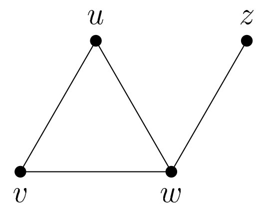
Un **graphe** $G = (V\,;E)$ est la donnée :

- d'un ensemble fini V dont les éléments sont appelés les **sommets** (*vertices* en anglais) de G ;
- d'une famille E de paires non ordonnées de sommets de V (non nécessairement distincts) appelés les **arêtes** (*edges* en anglais) de G.

On utilise le terme « famille » pour signifier une collection d'éléments dont certains peuvent apparaître plusieurs fois. Par exemple, $\{a\,;b\,;c\}$ est un ensemble, mais $\{a\,;a\,;b\,;c\,;a\,;c\}$ ne l'est pas ; c'est une famille dans le sens ci-dessus.

Exemples

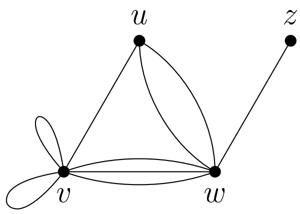
1)



$V = \{u\,;v\,;w\,;z\}$

$E = \{uv\,;uw\,;vw\,;wz\}$

2)

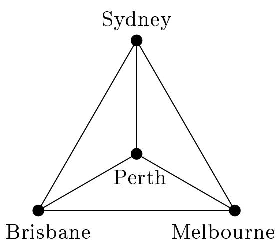


$V = \{u\,;v\,;w\,;z\}$

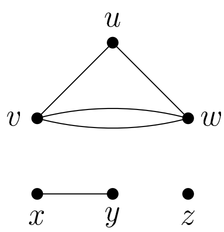
$E = \{uv\,;vv\,;vv\,;vw\,;vw\,;vw\,;uw\,;uw\,;wz\}$

2.1 Écrire l'ensemble des sommets et la famille des arêtes des graphes suivants :

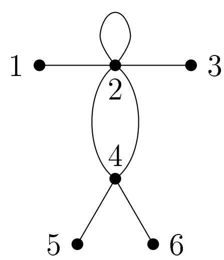
1)



2)



3)



2.2 Dessiner les graphes suivants :

1) $V = \{\square\,;\bigcirc\,;\Diamond\,;\triangle\}$ $\quad E = \{\square\bigcirc\,;\bigcirc\Diamond\,;\bigcirc\triangle\,;\Diamond\triangle\}$

2) $V = \{A\,;B\,;C\,;D\}$ $\quad E = \varnothing$

3) $V = \{1\,;2\,;3\,;4\,;5\,;6\,;7\,;8\}$ $\quad E = \{12\,;22\,;23\,;34\,;35\,;67\,;68\,;78\}$

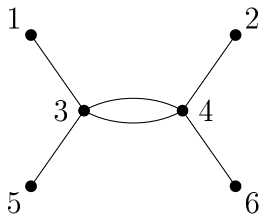
Deux sommets u et v d'un graphe sont dits **voisins** si uv est une arête de G.

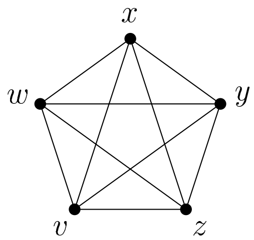
Un graphe est dit **simple** s'il ne contient ni arêtes multiples reliant deux sommets, ni boucles.

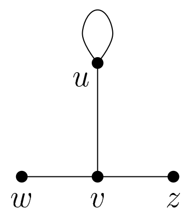
Ainsi, le graphe de l'exemple 1) est simple, alors que celui de l'exemple 2) ne l'est pas.

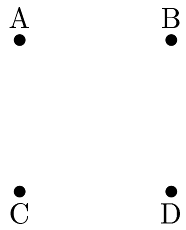
On appelle **ordre** d'un graphe $G = (V\,;E)$ le nombre de ses sommets ; on le note $|V|$.

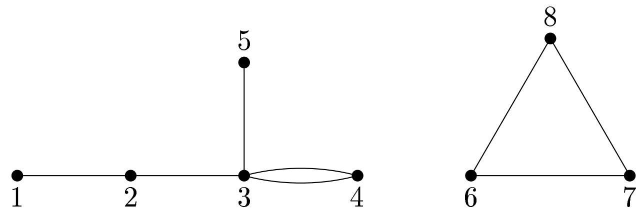
2.3 On considère les graphes suivants :

1) 

2) 

3) 

4) 

5) 

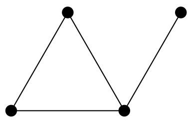
Parmi ces graphes, lesquels

1) contiennent des arêtes multiples ?
2) contiennent une boucle ?
3) sont simples ?

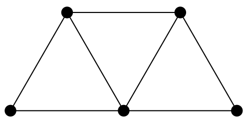
Un **sous-graphe** G′ d'un graphe G est un graphe tel que chaque sommet et chaque arête de G′ soient un sommet et une arête de G.

Exemple

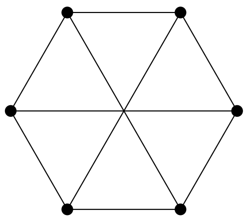
G′



G⋆

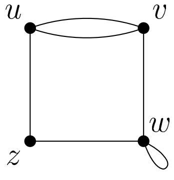


G⋆⋆

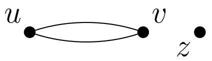


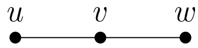
G′ est un sous-graphe de G⋆, mais pas de G⋆⋆.

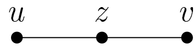
2.4 Soit G le graphe étiqueté suivant :

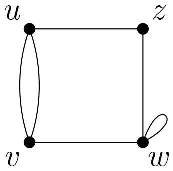


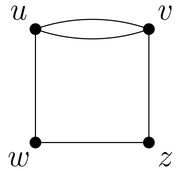
Lesquels de ces graphes sont des sous-graphes de G ?

1) 

2) 

3) 

4) 

5) 

Degré des sommets

Soit $G = (V\,;E)$ un graphe. Le **degré** d'un sommet $x \in V$ est le nombre d'arêtes de G contenant x ; on le note $\deg(x)$.

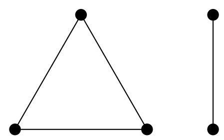
On convient que, lorsqu'il y a une boucle, elle contribue pour 2 (plutôt que 1) au degré de x.

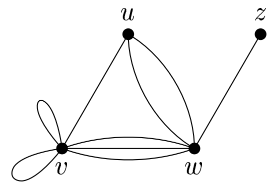
Un sommet de degré 0 est dit **isolé** et un sommet de degré 1 une **extrémité** du graphe.

Soit $G = (V\,;E)$ un graphe d'ordre n. À chaque sommet x_k, on peut faire correspondre son degré d_k. Quitte à renuméroter les sommets de G, on peut toujours les ordonner de façon à ce que la suite des degrés correspondants soit dans un ordre décroissant. La suite $(d_1, \ldots, d_n)$ ordonnée par ordre décroissant s'appelle la **liste des degrés** du graphe G. On dit alors que le graphe G est de **type** $(d_1, \ldots, d_n)$.

Exemples

1)

2) 

3) 

Dans les exemples 1) et 2) ci-dessus, chacun des graphes a deux sommets de degré 1 et trois sommets de degrés 2. Dans l'exemple 3), il y a un sommet de degré 1, un de degré 3, un de degré 6 et un de degré 8.

La liste des degrés des deux premiers graphes est donc $(2, 2, 2, 1, 1)$ et celle du troisième graphe est $(8, 6, 3, 1)$.

2.5 Pour chacun des graphes de l'exercice 2.3, donner

1) les degrés de tous les sommets ;
2) la liste des degrés.

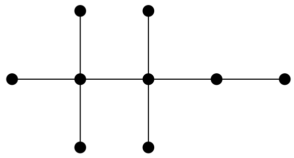
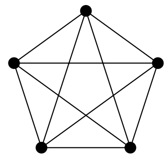
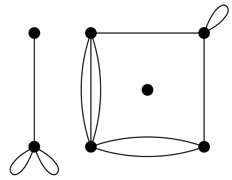
Lemme des poignées de mains

La somme des degrés de tous les sommets d'un graphe $G = (V\,;E)$ *est égale à deux fois le nombre de ses arêtes :*

$$\sum_{x \in V} \deg(x) = 2\,|E|$$

Preuve Chaque arête du graphe incrémente de deux la somme des degrés.

2.6 Écrire la liste des degrés de chacun des graphes suivants :

1) 
2) 
3) 

Vérifier le lemme des poignées de mains pour chacun de ces graphes.

2.7 Est-il possible de relier 15 ordinateurs de sorte que chaque appareil soit relié avec exactement trois autres ?

Indication : appliquer le lemme des poignées de mains.

2.8 Montrer que dans un graphe, le nombre de sommets de degré impair est pair.

Indication : Soit $\mathrm{G} = (\mathrm{V}\,;\mathrm{E})$ un graphe. Notons P l'ensemble des sommets de degré pair et I l'ensemble des sommets de degré impair. Alors $\sum\limits_{x \in \mathrm{V}} \deg(x) = \sum\limits_{x \in \mathrm{P}} \deg(x) + \sum\limits_{x \in \mathrm{I}} \deg(x)$.

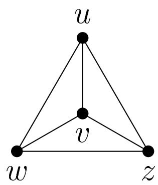
2.9

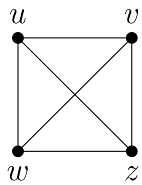
1) Construire un graphe tel que sa liste des degrés soit $(4, 3, 2, 1)$.
2) Est-il possible de construire un graphe tel que sa liste des degrés soit $(5, 4, 3, 2, 1)$?

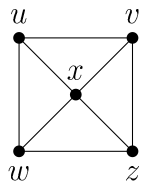
Isomorphisme de graphes

2.10 Dans chacune des lignes suivantes, deux des graphes sont identiques et le troisième est différent. Identifier l'intrus.

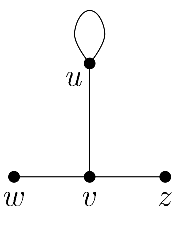
1)

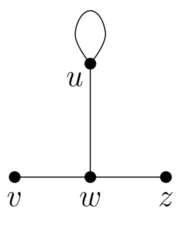


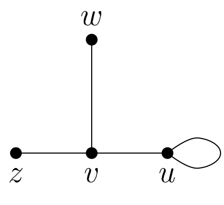




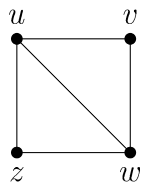
2)

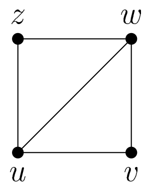


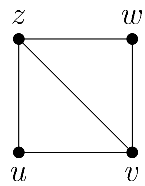




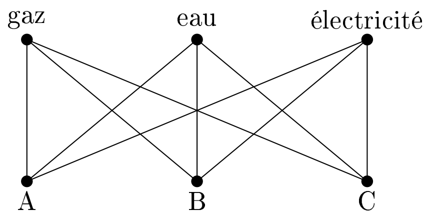
3)

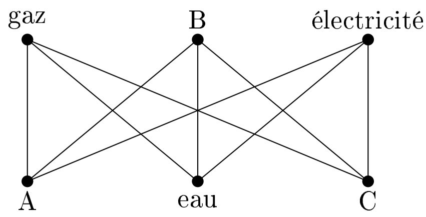






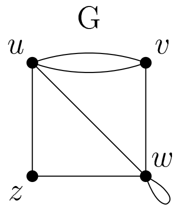
Considérons les deux graphes suivants :

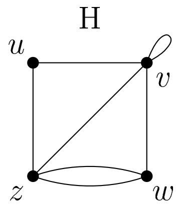




Ces deux graphes ne sont pas identiques, puisque le gaz et l'eau sont reliés dans le second, mais pas dans le premier. Pourtant, ils ont l'air très semblables : en permutant eau avec la lettre B, on passe d'un graphe à l'autre.

De la même manière, dans les graphes suivants :





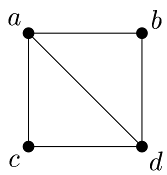
on peut re-étiqueter les sommets de G pour obtenir le graphe H.

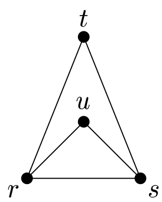
$$\begin{aligned} \mathrm{G} &\longrightarrow \mathrm{H} \\ u &\longmapsto z \\ v &\longmapsto w \\ w &\longmapsto v \\ z &\longmapsto u \end{aligned}$$

Notons que
– les deux arêtes uv dans G correspondent aux deux arêtes zw dans H ;
– l'arête uw dans G correspond à l'arête zv dans H ;
– la boucle en w dans G correspond à la boucle en v dans H ;
etc.

Deux graphes étiquetés G et H sont dits **isomorphes** [1] s'il existe une bijection entre les sommets de G et ceux de H de telle sorte que le nombre d'arêtes joignant chaque paire de sommets de G soit égal au nombre d'arêtes joignant les paires correspondantes de sommets de H.

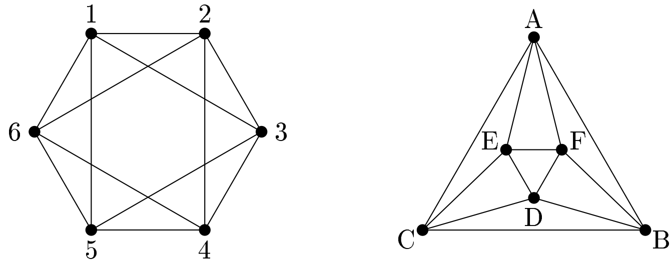
2.11 En explicitant une bijection entre les sommets, montrer que les graphes suivants sont isomorphes.



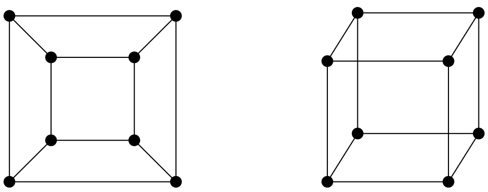


1. Ce terme est construit à partir de deux mots grecs : ἴσος et μορφή qui signifient « même » et « forme ». Il veut donc dire avoir la même forme.

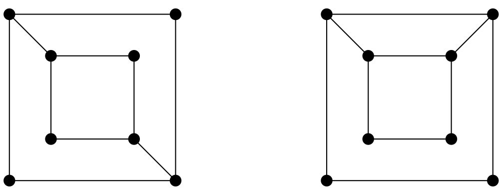
2.12 En explicitant une bijection entre les sommets, montrer que les graphes suivants sont isomorphes.



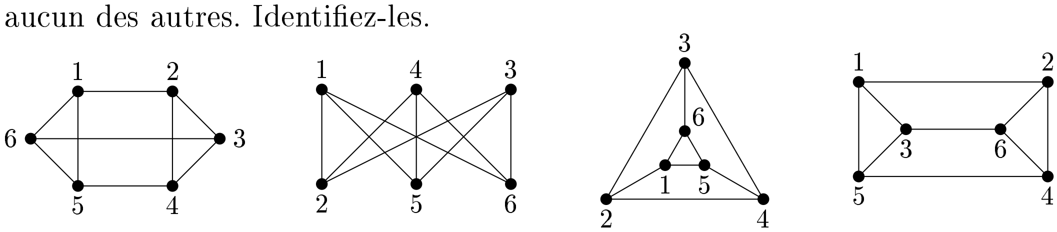
2.13 En étiquetant convenablement les sommets, montrer que les graphes suivants sont isomorphes.



2.14 Expliquer pourquoi les deux graphes ne sont pas isomorphes.



2.15 Parmi les quatre graphes étiquetés suivants, il y en a deux qui sont identiques, un qui est isomorphe aux deux précédents et le dernier qui n'est pas isomorphe à aucun des autres. Identifiez-les.

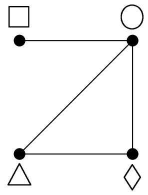


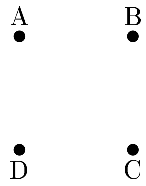
Réponses

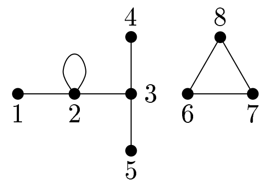
2.1 1) Posons B = Brisbane, M = Melbourne, P = Perth et S = Sydney.
$V = \{B\,;M\,;P\,;S\}$ $\quad E = \{BM\,;BP\,;BS\,;MP\,;MS\,;PS\}$

2) $V = \{u\,;v\,;w\,;x\,;y\,;z\}$ $\quad E = \{uv\,;uw\,;vw\,;vw\,;xy\}$

3) $V = \{1\,;2\,;3\,;4\,;5\,;6\}$ $\quad E = \{12\,;22\,;23\,;24\,;24\,;45\,;46\}$

2.2 1) 

2) 

3) 

2.3 1) graphes 1) et 5) 2) graphe 3) 3) graphes 2) et 4)

2.4 graphes 1), 2) et 4)

2.5 1) $\deg(1) = \deg(2) = \deg(5) = \deg(6) = 1$, $\deg(3) = \deg(4) = 4$
$(4, 4, 1, 1, 1, 1)$

2) $\deg(v) = \deg(w) = \deg(x) = \deg(y) = \deg(z) = 4$
$(4, 4, 4, 4, 4)$

3) $\deg(u) = \deg(v) = 3$, $\deg(w) = \deg(z) = 1$
$(3, 3, 1, 1)$

4) $\deg(A) = \deg(B) = \deg(C) = \deg(D) = 0$
$(0, 0, 0, 0)$

5) $\deg(1) = \deg(5) = 1$, $\deg(2) = \deg(4) = \deg(6) = \deg(7) = \deg(8) = 2$, $\deg(3) = 4$ $\quad (4, 2, 2, 2, 2, 2, 1, 1)$

2.6 1) $(4, 4, 2, 1, 1, 1, 1, 1, 1)$ $\quad 4 + 4 + 2 + 1 + 1 + 1 + 1 + 1 + 1 = 2 \cdot 8$

2) $(4, 4, 4, 4, 4)$ $\quad 4 + 4 + 4 + 4 + 4 = 2 \cdot 10$

3) $(5, 5, 4, 4, 3, 1, 0)$ $\quad 5 + 5 + 4 + 4 + 3 + 1 + 0 = 2 \cdot 11$

2.7 non

2.10 1) 3^{e} graphe 2) 2^{e} graphe 3) 3^{e} graphe

2.11
$a \longmapsto r$
$b \longmapsto t$
$c \longmapsto u$
$d \longmapsto s$

2.12
$1 \longmapsto A$
$2 \longmapsto B$
$3 \longmapsto C$
$4 \longmapsto D$
$5 \longmapsto E$
$6 \longmapsto F$

2.15 Les premier et troisième graphes sont identiques ; le deuxième graphe n'est pas isomorphe aux autres.

3 Chemins & Arbres

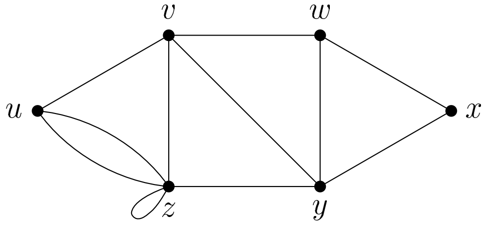
Chemins

Chaînes

Dans un graphe G, une **chaîne** allant de a à b est une liste ordonnée $x_0\,x_1\,\ldots\,x_{n-1}\,x_n$ de $n+1$ sommets de G où $a = x_0$, $b = x_n$ et où chaque paire $x_{i-1}\,x_i$ $(1 \leqslant i \leqslant n)$ est une arête de G.

Le nombre n des arêtes qui composent la chaîne est sa **longueur**.

Remarque On ne demande pas que dans une chaîne tous les sommets ou toutes les arêtes soient différents.



Dans le graphe ci-dessus, $u\,v\,w\,x\,y\,w\,v\,z\,z\,y$ est une chaîne de longueur 9 qui va du sommet u au sommet y. L'arête $v\,w$ est incluse deux fois, de même que les sommets v, w, y et z.

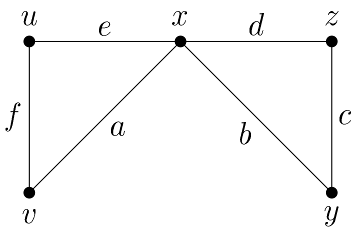
Si $a = b$, on parle d'une chaîne **fermée**, sinon d'une chaîne **ouverte**.

Chemins

Un **chemin** est une chaîne telle que chaque arête de celle-ci soit parcourue une seule fois.

Un **chemin simple** est un chemin dont chaque sommet est traversé une seule fois (excepté peut-être le premier et le dernier).

3.1 Considérons le graphe suivant :



Les chaînes $a\,e\,f\,a\,d$ et $a\,b\,c\,d\,e$ sont-elles des chemins ? des chemins simples ?

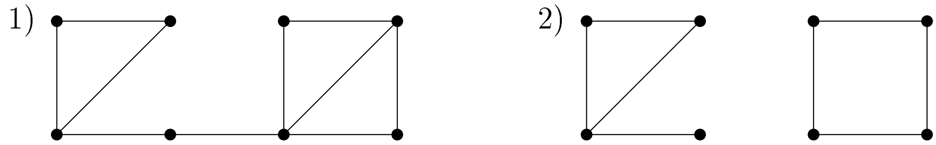
Remarque Comme l'illustre l'exercice 3.1, un chemin peut passer plusieurs fois par le même sommet.

Il est facile de montrer que tout chemin allant d'un sommet à un autre peut être « simplifié » en un chemin simple. Pour cela, il suffit de supprimer les détours.

Connexité

Un graphe est **connexe** si toute paire de sommets peut être reliée par un chemin.

3.2 Les graphes suivants sont-ils connexes ?



Si un graphe G n'est pas connexe, il se décompose en réunion de sous-graphes connexes, appelés **composantes connexes** de G.

Le second graphe de l'exercice 3.2 admet, par exemple, deux composantes connexes.

Théorème *Tout graphe connexe à n sommets possède au moins $n-1$ arêtes.*

Démontrons le résultat par récurrence sur le nombre de sommets n.

Pour $n = 1$ ou $n = 2$, le résultat est évident.

Supposons à présent $n \geqslant 3$ et le résultat vrai pour les graphes d'ordre $\leqslant n-1$.

Soit $\mathrm{G} = (\mathrm{V}\,;\mathrm{E})$ un graphe connexe d'ordre n. Distinguons deux cas.

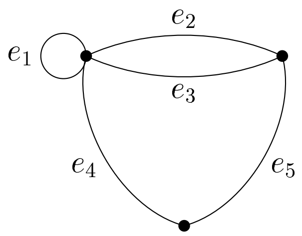
1) Supposons qu'il existe un sommet de degré 1.
   Soit G' le sous-graphe de G obtenu par suppression d'un sommet de degré 1 et de l'arête adjacente à ce sommet. Alors G' est un graphe connexe avec $n-1$ sommets. Vu l'hypothèse de récurrence, il possède au moins $(n-1)-1$ arêtes. Il en résulte que G, qui possède une arête supplémentaire, a au moins $n-1$ arêtes.
2) Supposons qu'il n'existe pas de sommet de degré 1.
   Vu la connexité de G, il ne peut pas y avoir de sommet isolé, de sorte que tous les sommets sont de degré $\geqslant 2$.
   Le lemme des poignées de mains implique $2\,|\mathrm{E}| = \sum_{x \in \mathrm{V}} \deg(x) \geqslant 2\,n$, d'où l'on conclut que le nombre d'arêtes $|\mathrm{E}| \geqslant n \geqslant n-1$.

Cycles

Un **cycle** est un chemin simple fermé.

Un graphe ne contenant pas de cycle est **acyclique**.

3.3 Déterminer les cycles de longueur 1, 2 et 3 dans le graphe suivant :



Proposition *Si dans un graphe* G *tout sommet est de degré* $\geqslant 2$*, alors* G *possède au moins un cycle.*

Preuve La preuve utilise un algorithme de marquage. Initialement, tous les sommets sont non marqués. Un sommet x_1 est arbitrairement marqué.

L'algorithme construit une séquence $x_1, x_2, \dots, x_k$ de sommets marqués en choisissant arbitrairement pour x_{i+1} un sommet non marqué adjacent à x_i.

L'algorithme s'arrête lorsque x_k ne possède plus de voisin non marqué. Puisque ce sommet est degré $\geqslant 2$, il possède, outre x_{k-1}, un autre voisin marqué x_j.

Alors $x_k x_j x_{j+1} \dots x_{k-1} x_k$ est un cycle.

Corollaire *Un graphe acyclique possède au moins un sommet de degré* $\leqslant 1$.

Théorème *Tout graphe acyclique à* n *sommets possède au plus* $n-1$ *arêtes.*

Preuve Démontrons le résultat par récurrence sur le nombre de sommets n.

Pour $n = 1$ ou $n = 2$, le résultat est évident.

Supposons à présent $n \geqslant 3$ et le résultat vrai pour les graphes d'ordre $\leqslant n-1$.

Soit $\mathrm{G} = (\mathrm{V}\,;\mathrm{E})$ un graphe acyclique d'ordre n. D'après le corollaire, il existe un sommet x de degré $\leqslant 1$. Soit G' le sous-graphe de G obtenu par suppression du sommet x et de l'éventuelle arête adjacente à ce sommet. Alors G' est un graphe acyclique avec $n-1$ sommets. Vu l'hypothèse de récurrence, il possède au plus $(n-1)-1$ arêtes. Il en résulte que G, qui possède une éventuelle arête supplémentaire, a au plus $n-1$ arêtes.

Arbres

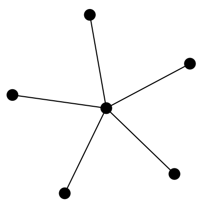
Un **arbre** est un graphe connexe acyclique.

3.4 Quel est le nombre d'arêtes d'un arbre à n sommets ?

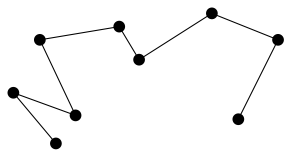
Remarque Un arbre est nécessairement simple, puisqu'il est acyclique.

Exemple Les trois graphes suivants sont des arbres :

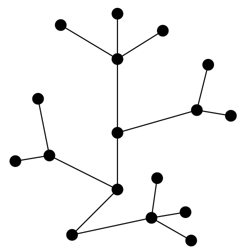
1)



2)



3)



La réunion ensembliste des graphes 1), 2) et 3) est un graphe qui, bien que n'admettant pas de cycles, n'est pas un arbre, car il n'est pas connexe. On l'appelle une **forêt**.

Remarque Un sous-graphe d'un arbre peut être une forêt ; un sous-graphe connexe d'un arbre T est un **sous-arbre** de T.

Propriétés & caractérisations des arbres

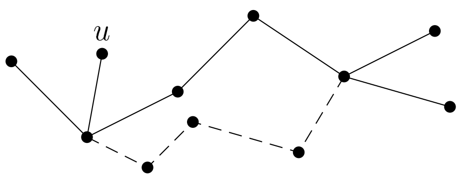
La connexité d'un graphe implique que deux sommets quelconques sont toujours reliés par *au moins un* chemin.

Le fait qu'il n'y ait qu'*un seul* chemin entre deux sommets distincts quelconques caractérise les arbres, comme l'énonce le théorème suivant.

Théorème *Un graphe simple est un arbre si et seulement si deux sommets distincts quelconques peuvent être reliés par un chemin unique.*

Preuve

1) Soit T un arbre. Supposons — par l'absurde — qu'il existe deux sommets distincts u et v qui puissent être reliés par deux chemins distincts.



Leur réunion va contenir un cycle (et probablement d'autres arêtes), ce qui est en contradiction avec le fait que T est acyclique.

2) Réciproquement, supposons que deux sommets distincts quelconques d'un graphe G soient toujours reliés par un chemin unique. Alors G ne peut pas contenir de cycle, car deux sommets distincts d'un cycle sont toujours reliés par deux chemins distincts. Donc G est un arbre.

Théorème *Soit* G *un graphe à* n *sommets. Alors les propriétés suivantes sont équivalentes et peuvent être prises comme la définition d'un arbre.*

1) G *est connexe et acyclique.*

2) G *est un graphe connexe à* $n-1$ *arêtes.*

3) G *est connexe et la suppression de toute arête le déconnecte.*

4) G *est un graphe acyclique à* $n-1$ *arêtes.*

5) G *est acyclique et l'ajout de toute arête le rend cyclique.*

Preuve

1) $\Rightarrow$ 2) G étant connexe et acyclique, il possède exactement $n-1$ arêtes.

2) $\Rightarrow$ 3) Pour être connexe, un graphe à n sommets doit posséder au moins $n-1$ arêtes. En supprimant une arête de G, il n'en reste plus que $n-2$.

3) $\Rightarrow$ 4) Si, par l'absurde, G possédait un cyclique, la suppression d'une arête ne saurait le déconnecter ; par suite, G est acyclique. Puisqu'il est également connexe, il possède donc exactement $n-1$ arêtes.

4) $\Rightarrow$ 5) Pour être acyclique, un graphe à n sommets doit posséder au plus $n-1$ arêtes. L'ajout d'une arête à G donne un graphe à n arêtes.

5) $\Rightarrow$ 1) Considérons deux sommets x et y de G.

- Si l'arête $x\,y$ existe, alors c'est un chemin de x à y.
- Sinon, ajoutons l'arête $x\,y$ à G : nous créons alors un cycle de la forme $x\,a\,\ldots\,w\,y\,x$. Ceci montre l'existence du chemin $x\,a\,\ldots\,w\,y$ entre x et y dans G.

G est donc bien un graphe connexe.

3.5 Montrer que les conditions suivantes sont équivalentes :

1) G est connexe et a un seul cycle.

2) G est connexe et le nombre de sommets est égal au nombre d'arêtes.

3) Il existe une arête e de G telle que $G-e$ est un arbre.

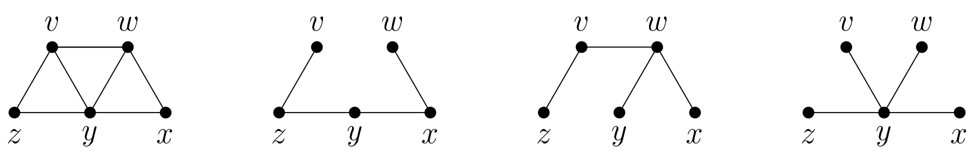
3.6 Prouver qu'un graphe de n sommets et $n-1$ arêtes qui a au moins un cycle a plus d'une composante connexe.

Connexion minimale

Arbres de recouvrement d'un graphe

On appelle **arbre de recouvrement** d'un graphe G un sous-graphe de G qui contient tous les sommets de G et qui est un arbre.

En général, comme on le voit ci-dessous, un graphe peut avoir plusieurs arbres de recouvrement.



Remarque Si le graphe n'est pas connexe, il n'existe pas d'arbre de recouvrement, puisqu'un arbre est connexe.

Théorème *Tout graphe connexe contient un arbre de recouvrement.*

Preuve Soit G un graphe connexe.

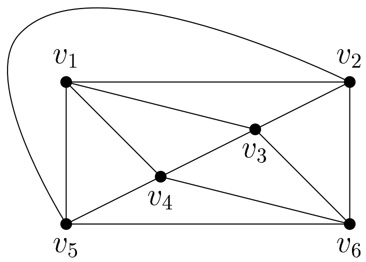
Considérons l'ensemble Ψ de tous les sous-graphes connexes de G contenant tous les sommets de G. L'ensemble Ψ est non vide, vu que $G \in \Psi$.

Soit T l'un des éléments de Ψ possédant un nombre minimum d'arêtes.

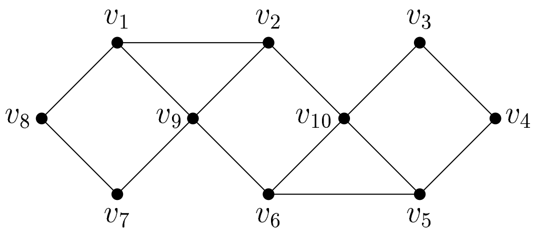
T est acyclique : sinon, T contiendrait un cycle et la suppression d'une arête quelconque de ce cycle donnerait un sous-graphe appartenant à Ψ ayant une arête de moins que T, ce qui contredirait le choix de T.

Vu que T est connexe et acyclique, il constitue un arbre de recouvrement de G.

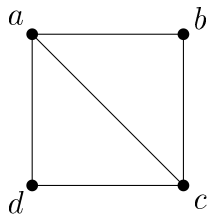
3.7 Trouver un arbre de recouvrement du graphe suivant :



3.8 Trouver un arbre de recouvrement du graphe suivant :



3.9 Dessiner les 8 arbres de recouvrement du graphe :



Le problème de la connexion minimale

Supposons que différents objets (villes, centres de distribution, prises de courant électrique, etc.) doivent être reliés entre eux de manière minimum (cela peut être en distance, en temps, en coût ou selon d'autres critères), de sorte qu'il existe toujours un « chemin » possible entre deux quelconques de ces objets.

Il est facile de traduire cette situation par un graphe G :

– L'ensemble des sommets de G est l'ensemble des objets. Une arête de G correspond à une liaison directe possible entre deux sommets.
– À chaque arête on fait correspondre un nombre positif, appelé **poids**, qui peut représenter une distance, un temps, un coût, etc.

Nous obtenons ainsi ce que l'on appelle un **graphe pondéré**.

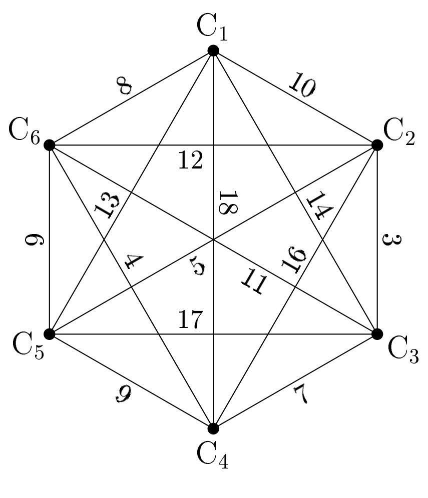
Le problème consiste alors à trouver un arbre de recouvrement de G de poids minimum.

Exemple Six ordinateurs $C_1, C_2, \ldots, C_6$ doivent être reliés par un réseau de transmission à fibre optique. Le coût (unité = 10 000 fr.) de chaque liaison possible est donné par le tableau suivant :

	C_1	C_2	C_3	C_4	C_5	C_6
C_1	—	10	14	18	13	8
C_2	10	—	3	16	5	12
C_3	14	3	—	7	17	11
C_4	18	16	7	—	9	4
C_5	13	5	17	9	—	6
C_6	8	12	11	4	6	—

Il faut trouver le réseau le moins cher possible tel que toute paire d'ordinateurs puisse communiquer, que ce soit directement ou à travers d'autres ordinateurs.

Le graphe pondéré par les coûts est le suivant :



Algorithme de Kruskal

Cet algorithme est dû au mathématicien tchèque Joseph B. Kruskal qui l'a utilisé le premier en 1956.

Soit G = (V ; E) un graphe connexe et pondéré.

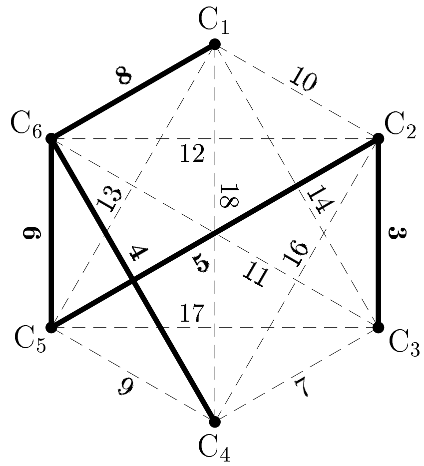
1) Trier les arêtes de E par ordre croissant de poids : $e_1, \dots, e_{|E|}$.
2) Poser F = ∅.
3) Pour tout i allant de 1 à n, ajouter l'arête e_i à F, pour autant que le graphe (V ; F) qui en résulte demeure acyclique.

Par construction et en vertu de la propriété 5) du théorème de la page 3.4, on obtient ainsi un arbre de recouvrement de poids minimum.

Exemple (suite) Appliquons l'algorithme de Kruskal aux six ordinateurs reliés par un réseau de transmission à fibre optique.

Ordonnons les arêtes du graphe par ordre croissant de poids :
$3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 16, 17, 18$.

L'application de l'algorithme donne $F = \{3 ; 4 ; 5 ; 6 ; 8\}$.



En effet, l'ajout de l'arête de poids 7 créerait le cycle $C_2\, C_3\, C_4\, C_6\, C_5\, C_2$. À partir de l'ajout de l'arête de poids 8, tous les sommets sont reliés, c'est-à-dire que le graphe est connexe, de sorte qu'il constitue un arbre et que l'ajout de toute arête supplémentaire crée un cycle.

Le poids de l'arbre est $3 + 4 + 5 + 6 + 8 = 26$: le réseau cherché peut être construit, selon l'arbre trouvé, au prix de 260 000 fr.

Algorithme de Prim

Bien que l'algorithme de Kruskal puisse être facilement appliqué « à la main » quand le graphe est petit, il n'est pas particulièrement approprié à une implémentation efficace dans un ordinateur. En effet, il faut arranger les arêtes dans l'ordre de poids croissant et surtout contrôler qu'aucun cycle n'a été créé.

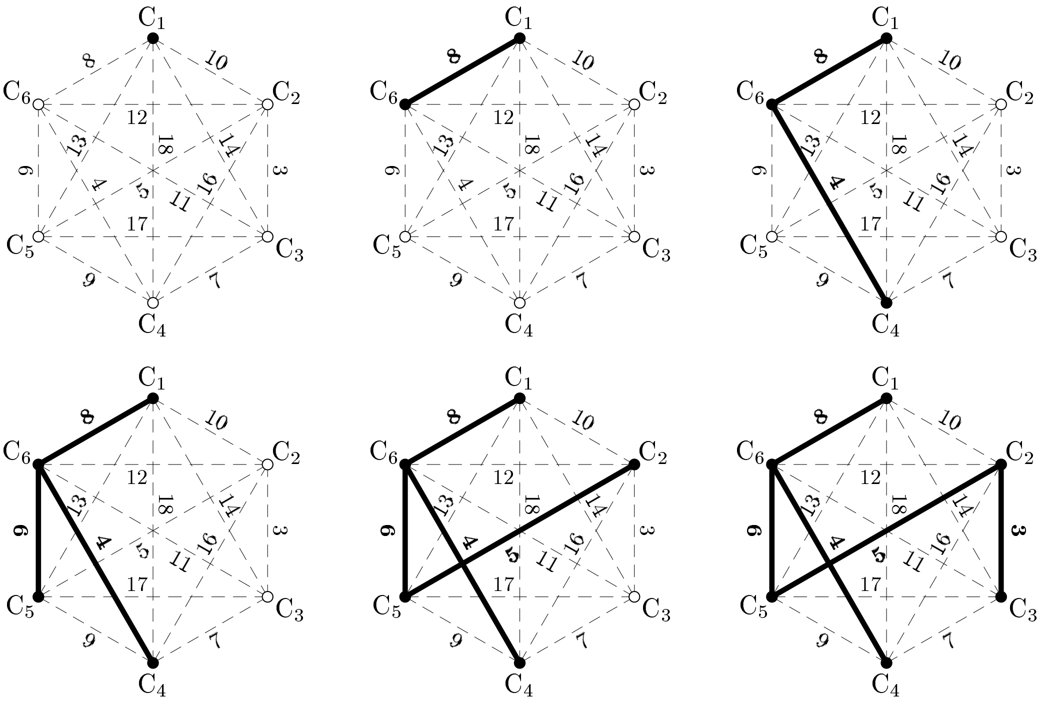
L'algorithme de Prim, que l'on doit à l'informaticien Robert C. Prim en 1957, permet de surmonter ces difficultés.

Soit $G = (V\,;E)$ un graphe connexe et pondéré.

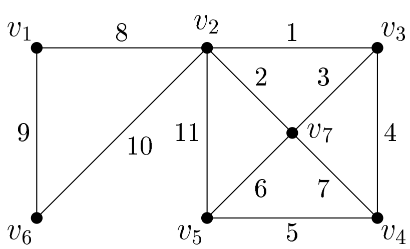
1) Marquer arbitrairement un sommet.
2) Poser $F = \varnothing$.
3) Tant qu'il existe un sommet non marqué, choisir une arête de poids minimum joignant un sommet marqué à un sommet non marqué ; ajouter cette arête à F et marquer ce nouveau sommet.

Par construction et en vertu de la propriété 3) du théorème de la page 3.4, on obtient ainsi un arbre de recouvrement de poids minimum.

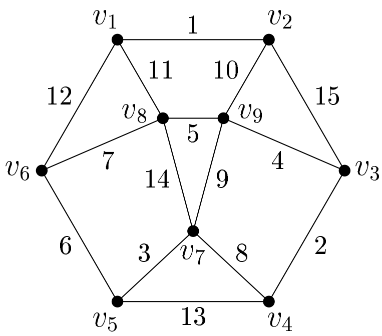
Exemple (fin) Appliquons l'algorithme de Prim aux six ordinateurs reliés par un réseau de transmission à fibre optique, en marquant initialement le sommet C_1 :



3.10 Utiliser les algorithmes de Kruskal et de Prim pour trouver un arbre de recouvrement minimum pour le graphe pondéré suivant :



3.11 Utiliser les algorithmes de Kruskal et de Prim pour trouver un arbre de recouvrement minimum pour le graphe pondéré suivant :



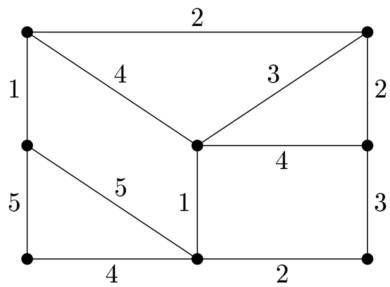
3.12 Le tableau suivant donne les distances (en centaines de milles) entre six villes européennes.

	Berlin	Londres	Madrid	Moscou	Paris	Rome
Berlin	0	7	15	11	7	10
Londres	7	0	11	18	3	12
Madrid	15	11	0	27	8	13
Moscou	11	18	27	0	18	20
Paris	7	3	8	18	0	9
Rome	10	12	13	20	9	0

Trouver un arbre de recouvrement minimum reliant chacune de ces villes :

1) par l'algorithme de Kruskal ;
2) par l'algorithme de Prim.

3.13 Déterminer tous les graphes de recouvrement minimaux du graphe suivant :



Réponses

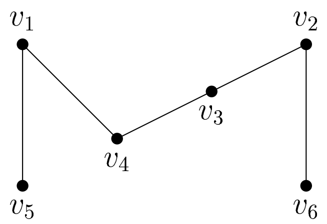
3.1 $a\,e\,f\,a\,d$ n'est pas un chemin allant de v à z.
$a\,b\,c\,d\,e$ est un chemin allant de v à u qui n'est pas simple.

3.2 1) oui 2) non

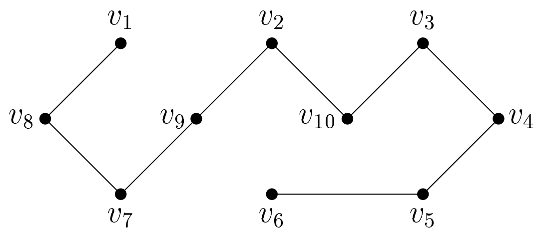
3.3 e_1 est le seul cycle de longueur 1.
e_2e_3 est le seul cycle de longueur 2.
$e_2e_4e_5$ et $e_3e_4e_5$ sont les seuls cycles de longueur 3.
Remarquons que $e_3e_2e_1$ n'est pas un cycle puisqu'un sommet est répété.

3.4 $n-1$

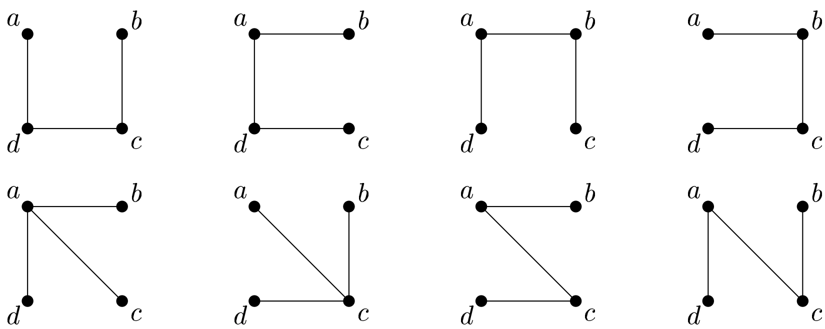
3.7



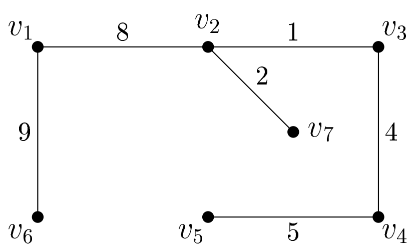
3.8



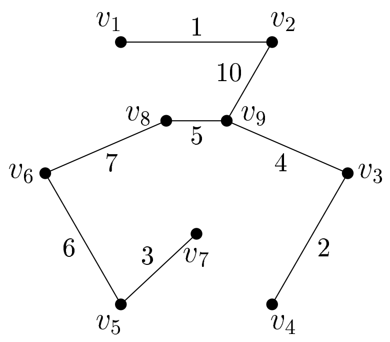
3.9



3.10

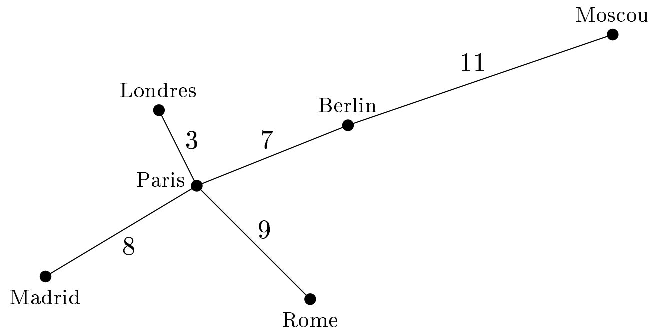


3.11



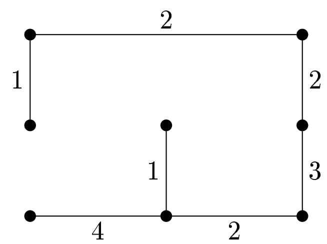
v_1
1
v_2
10
v_8
5
v_9
v_6
7
4
v_3
6
3
v_7
2
v_5
v_4

3.12

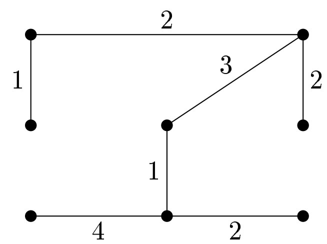


Moscou
11
Londres
Berlin
3
7
Paris
9
8
Madrid
Rome

3.13



2
1
2
1
3
4
2



2
3
1
2
1
4
2

4 Graphes eulériens

Problème de l'explorateur

Un explorateur souhaite explorer toutes les routes entre un certain nombre de villes. Peut-on trouver un itinéraire qui passe par chaque route une seule fois ?

Ce problème se traduit aisément dans le langage des graphes : peut-on trouver un chemin passant une et une seule fois par chaque arête d'un graphe ?

Étant donné que cette question est étroitement liée au problème des ponts de Könisberg et que Euler y est historiquement associé, on pose les définitions suivantes.

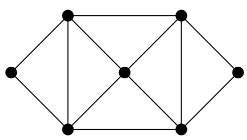
Un chemin d'un graphe G est appelé **chemin eulérien** s'il passe une et une seule fois par chaque arête du graphe.

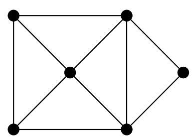
Un graphe G est un **graphe eulérien** s'il admet un chemin eulérien fermé.

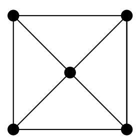
Un graphe G est **semi-eulérien** s'il n'est pas eulérien et s'il admet un chemin eulérien ouvert.

Remarque Un graphe eulérien ou semi-eulérien est nécessairement connexe.

Exemples Les trois graphes ci-dessous sont respectivement eulérien, semi-eulérien et non eulérien.

1)


2)


3)


Théorème *Un graphe est eulérien si et seulement si il est connexe et tous ses sommets sont de degré pair.*

Preuve Supposons qu'un graphe G soit eulérien. Il existe alors un chemin fermé c parcourant une et une seule fois chaque arête.

Le graphe G est donc connexe, puisque c relie tous les sommets entre eux.

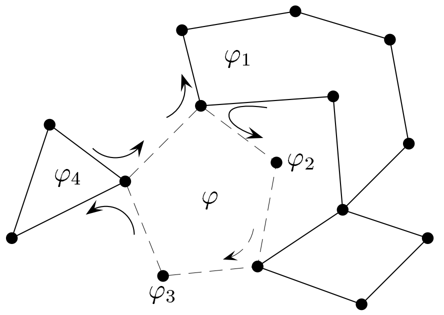
Considérons un sommet x. Lors du parcours du cycle, à chaque fois que nous passons par lui, nous y arrivons et nous en repartons par 2 arêtes non encore parcourues. Le sommet x est donc de degré pair.

Réciproquement, considérons un graphe G connexe dont tous les sommets sont de degré pair. Nous allons montrer par récurrence sur le nombre d'arêtes que G est alors eulérien.

Si G se réduit à un unique sommet isolé, il est évidemment eulérien.

Sinon tous les sommets de G sont de degré $\geqslant 2$. La proposition de la page 3.3 implique qu'il existe un cycle φ sur G.

Considérons le graphe partiel H constitué des arêtes en dehors du cycle φ. Les sommets de H sont également de degré pair, le cycle contenant un nombre pair d'arêtes incidentes pour chaque sommet. Par hypothèse de récurrence, chaque composante connexe H_i de H est un graphe eulérien et admet donc un chemin eulérien fermé φ_i.



Le cycle φ, représenté en trait tillé, définit 4 composantes connexes pour le graphe H, dont 2 sommets isolés pour lesquels leur cycle eulérien est sans arête.
Les flèches symbolisent l'opération de fusion des 2 cycles non vides avec φ.

Pour reconstruire un chemin eulérien fermé sur G, il nous suffit de fusionner le cycle φ avec les différents cycles φ_i. Pour cela, on parcourt le cycle φ depuis un sommet arbitraire ; lorsque l'on rencontre pour la première fois un sommet appartenant à H_i, on lui substitue le chemin eulérien fermé φ_i. Le chemin obtenu est un chemin eulérien fermé pour G, le cycle φ et les chemins φ_i formant une partition des arêtes.

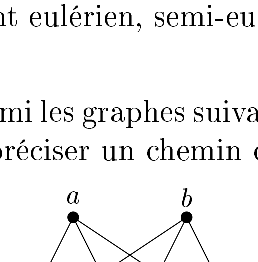
Corollaire *Un graphe est semi-eulérien si et seulement si il est connexe et s'il a exactement deux sommets de degré impair.*

Dans ce cas, le chemin eulérien ouvert joint ces deux sommets.

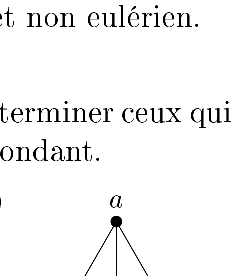
4.1 Justifier que les graphes des exemples de la page précédente sont respectivement eulérien, semi-eulérien et non eulérien.

4.2 Parmi les graphes suivants, déterminer ceux qui sont eulériens ou semi-eulériens et préciser un chemin correspondant.

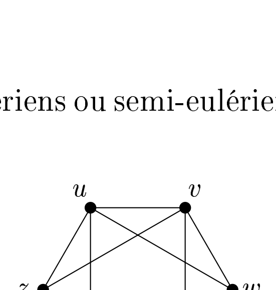
1)



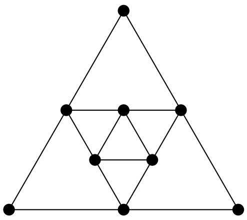
2)

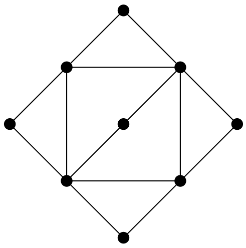


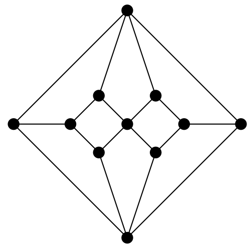
3)



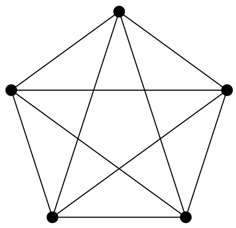
4.3 Déterminer si les graphes suivants sont eulériens, semi-eulériens ou ni l'un ni l'autre.

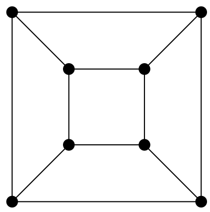
1) 

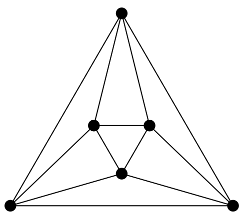
2) 

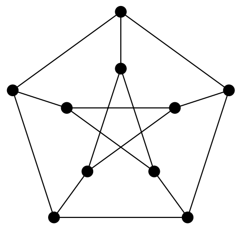
3) 

4.4 Déterminer si les graphes suivants sont eulériens, semi-eulériens ou ni l'un ni l'autre.

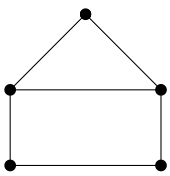
1) 

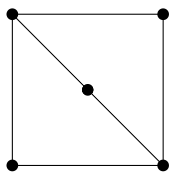
2) 

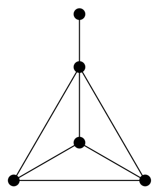
3) 

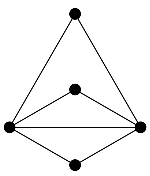
4) 

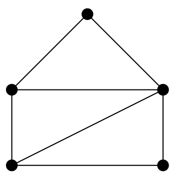
4.5 Déterminer si les graphes suivants sont eulériens, semi-eulériens ou ni l'un ni l'autre.

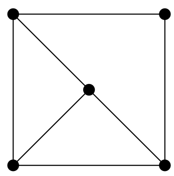
1) 

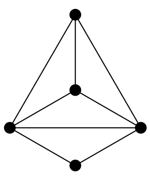
2) 

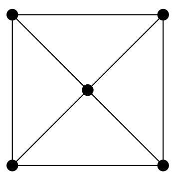
3) 

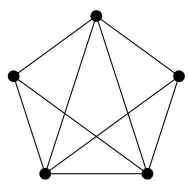
4) 

5) 

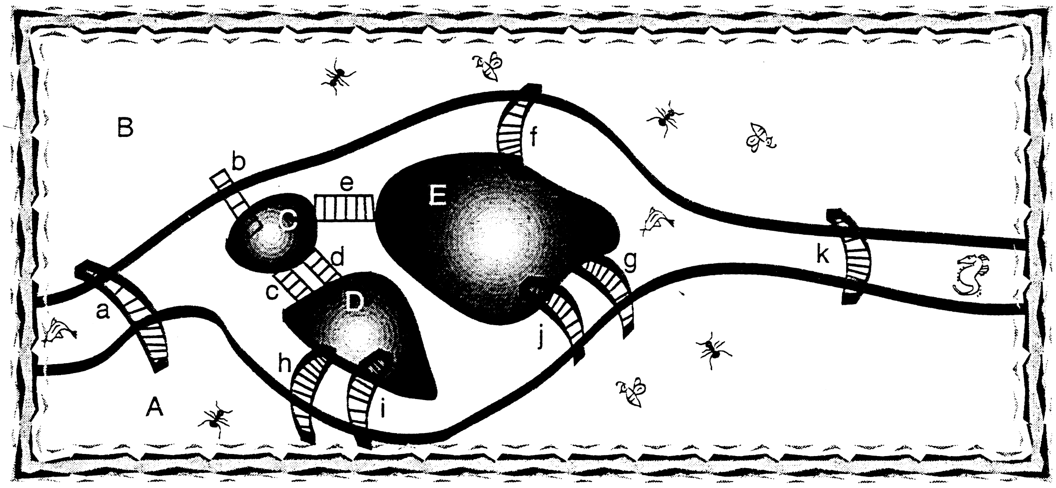
6) 

7) 

8) 

9) 

4.6 On représente ci-dessous la ville de Dreamtown avec sa rivière et ses trois îles.



Est-il possible de parcourir chaque pont une fois et une seule fois lors d'une balade dans cette ville ?

Algorithme de Fleury

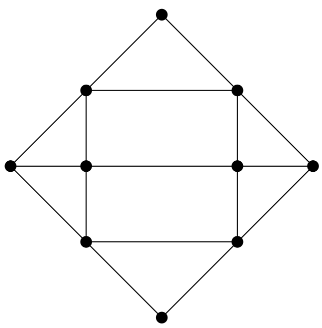
Il existe un algorithme pour déterminer les chemins eulériens dans un graphe eulérien. L'idée est de parcourir le graphe en supprimant toutes les arêtes traversées, mais en évitant autant que possible de rendre le graphe non connexe.

Une arête $a\,b$ d'un graphe G est appelée un **pont** si $a\,b$ est l'unique chemin entre les sommets a et b.

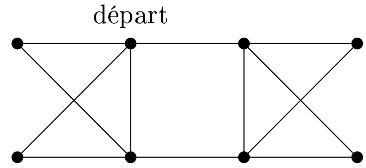
Dans un graphe eulérien, l'**algorithme de Fleury** permet toujours d'obtenir un chemin eulérien fermé :

- Commencer à partir de n'importe quel sommet et parcourir les arêtes arbitrairement en respectant les règles suivantes :
- Supprimer les arêtes parcourues ; au cas où apparaît un sommet isolé, supprimer ce sommet.
- À chaque étape, n'utiliser un pont que s'il n'y a pas d'autre alternative.

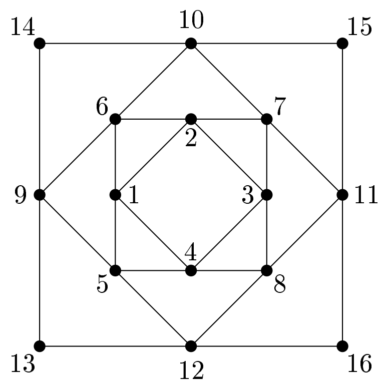
4.7 Utiliser l'algorithme de Fleury pour trouver un chemin eulérien ouvert dans le graphe suivant :



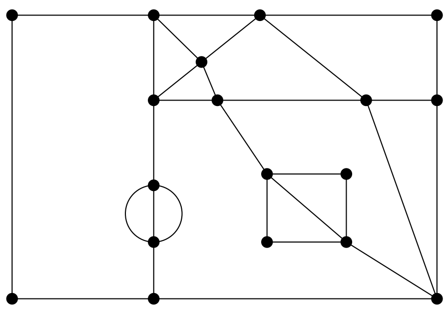
4.8 Utiliser l'algorithme de Fleury pour trouver au moins un chemin eulérien fermé dans le graphe suivant :



4.9 Utiliser l'algorithme de Fleury pour trouver au moins un chemin eulérien fermé dans le graphe suivant :



4.10 Vous êtes un agent de police et la carte des routes de votre secteur est représentée ci-dessous.



Est-il possible de patrouiller sur chacune de ces routes sans parcourir aucune d'elles plus d'une fois ?

Réponses

4.2 1) semi-eulérien $a\,c\,b\,d\,a\,e\,b$

2) ni eulérien, ni semi-eulérien

3) eulérien $u\,v\,z\,u\,y\,z\,x\,y\,w\,x\,v\,w\,u$

4.3 1) eulérien 2) semi-eulérien 3) ni eulérien, ni semi-eulérien

4.4 1) eulérien 2) ni eulérien, ni semi-eulérien

3) eulérien 4) ni eulérien, ni semi-eulérien

4.5 1) semi-eulérien 2) semi-eulérien

3) ni eulérien, ni semi-eulérien 4) eulérien

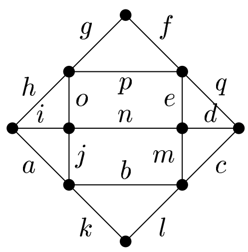
5) semi-eulérien 6) ni eulérien, ni semi-eulérien

7) semi-eulérien 8) ni eulérien, ni semi-eulérien

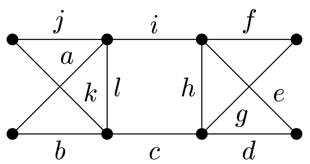
9) semi-eulérien

4.6 oui

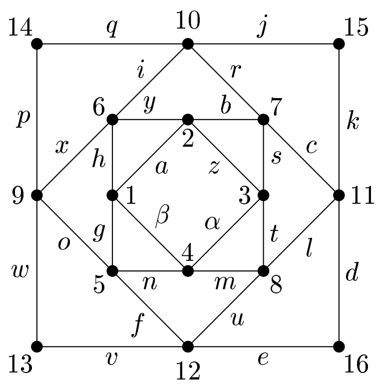
4.7



4.8



4.9



4.10 Oui, mais sans revenir au point de départ : graphe semi-eulérien

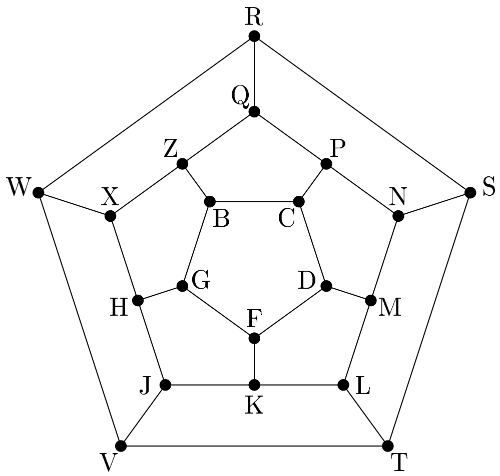
5 Graphes hamiltoniens

Problème du voyageur

Au chapitre précédent, nous avons examiné et résolu le problème de l'existence d'un chemin passant une et une seule fois par chacune des arêtes d'un graphe donné. Nous allons analyser le problème correspondant pour les sommets : dans un graphe donné, existe-t-il un chemin qui passe une et une seule fois par chacun de ses sommets ?

Nous retrouvons là la forme élémentaire du problème du voyageur de commerce. De manière surprenante, ce problème s'avère beaucoup plus difficile que celui de l'explorateur.

Vers 1850, le mathématicien William Hamilton (1805-1865) a tenté de populariser, malheureusement sans succès commercial, un casse-tête qu'il a appelé « the icosian game ». L'idée de ce jeu est illustrée dans le graphe ci-dessous.



Le graphe modélise l'itinéraire d'un voyage autour de la Terre où 20 villes doivent être visitées. Partant de l'une des vingt villes, le voyageur doit y revenir en ayant visité chaque ville une seule fois. Plus précisément, le jeu consiste à imposer les cinq premières villes à visiter et à demander au joueur de compléter l'itinéraire. Par exemple, si l'on commence par les villes BCPNM, on peut remarquer qu'il y a exactement deux voyages réalisant les conditions imposées :

BCPNM DFKLTS RQZXW VJHGB
BCPNM DFGHX WVJKLT SRQZB

Un **cycle hamiltonien** d'un graphe G est un cycle qui contient chaque sommet de G.

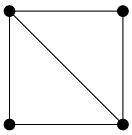
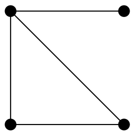
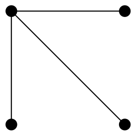
Un graphe est **hamiltonien** s'il contient un cycle hamiltonien.

Un **chemin hamiltonien** est un chemin simple ouvert qui contient chaque sommet de G.

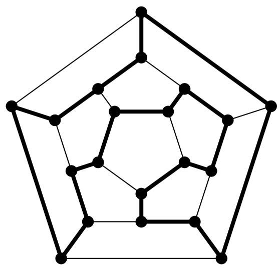
Un graphe non hamiltonien est **semi-hamiltonien** s'il contient un chemin hamiltonien.

Remarque Ajouter ou enlever des boucles ou des arêtes multiples ne modifie pas le caractère hamiltonien d'un graphe, puisque l'on peut les ignorer en cherchant un chemin qui visite chaque sommet.

Exemples Les trois graphes ci-dessous sont respectivement hamiltonien, semi-hamiltonien et non hamiltonien.

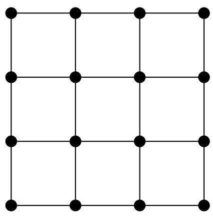
1)  2)  3) 

Le graphe ci-dessous est hamiltonien ; on le voit en considérant, par exemple, le cycle hamiltonien dessiné en gras :



5.1 Trouver un autre circuit hamiltonien dans le « icosian game ».

5.2 Montrer que le graphe ci-dessous est hamiltonien en exhibant un cycle hamiltonien.



À première vue, le problème de savoir si un graphe est hamiltonien semble très proche de celui de savoir si un graphe est eulérien. On pourrait espérer résoudre ce problème d'une manière aussi satisfaisante que pour les graphes eulériens. Dans le chapitre précédent, nous avons formulé un critère pour l'existence d'un chemin eulérien — le degré de chaque sommet doit être pair — et nous avons même proposé, lorsque cette condition est vérifiée, un algorithme permettant de construire le chemin cherché.

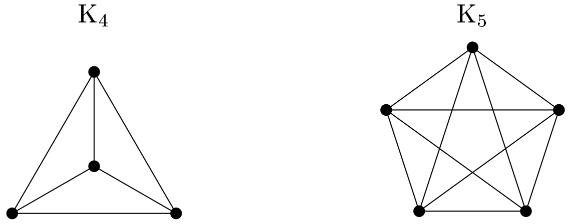
Malheureusement, la solution du problème correspondant pour un cycle hamiltonien est beaucoup plus difficile. En fait, dans le cas d'un graphe quelconque, il n'y a toujours pas de critères, c'est-à-dire de conditions nécessaires et suffisantes assurant l'existence d'un cycle ou d'un chemin hamiltonien. Cela reste l'un des problèmes majeurs non résolus de la théorie des graphes.

Cependant, nous possédons des résultats partiels de deux types. Les uns donnent des conditions nécessaires, c'est-à-dire des conditions que tout graphe doit satisfaire pour être hamiltonien ; les autres constituent des conditions suffisantes, c'est-à-dire des conditions qui assurent qu'un graphe est hamiltonien.

Conditions suffisantes pour qu'un graphe soit hamiltonien

Un graphe est **complet** lorsque deux sommets quelconques sont reliés par exactement une arête. Le graphe complet d'ordre n se note K_n.

Exemples



Trouver un cycle hamiltonien dans un graphe complet K_n ($n \geqslant 2$) est très simple. On peut procéder à partir de n'importe quel sommet. Comme toutes les arêtes possibles sont présentes, on peut toujours passer de n'importe quel sommet à n'importe quel autre directement. La construction se fait de proche en proche.

5.3
1) Quel est le nombre d'arêtes du graphe K_n ?
2) Combien de cycles hamiltoniens distincts y a-t-il dans K_n ?

La plupart des théorèmes qui expriment une condition suffisante pour qu'un graphe G soit hamiltonien sont de la forme : si G a « assez » d'arêtes, alors G est hamiltonien.

Théorème d'Ore (1960)

Soit G *un graphe simple avec* $n \geqslant 3$ *sommets. Si* $\deg(u) + \deg(v) \geqslant n$ *pour toute paire de sommets* u *et* v *non voisins, alors* G *est hamiltonien.*

Preuve Par l'absurde, supposons le théorème faux.

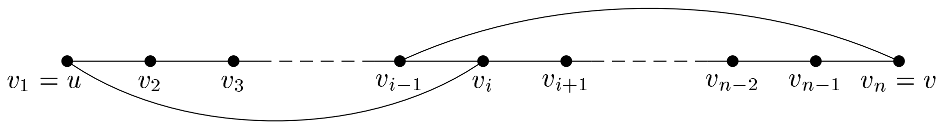
Il existe donc un graphe non hamiltonien de n sommets satisfaisant la condition d'Ore : $\deg(u) + \deg(v) \geqslant n$ pour toute paire de sommets u et v non voisins.

En ajoutant des arêtes supplémentaires — ce qui ne perturbe pas l'inégalité d'Ore — on peut trouver un nouveau graphe $\mathrm{G}^\star$ qui soit « à peine » non hamiltonien dans le sens suivant : l'adjonction d'une nouvelle arête le rendrait hamiltonien. Quitte à renuméroter les sommets de $\mathrm{G}^\star$, il s'ensuit que $\mathrm{G}^\star$ possède un chemin $u = v_1 \, \ldots \, v_n = v$ passant par chaque sommet.

Puisque $G^\star$ est non hamiltonien, les sommets u et v ne sont pas voisins et satisfont la condition d'Ore : $\deg(u) + \deg(v) \geqslant n$.

Désignons par $N(u)$ l'ensemble de tous les sommets voisins de u et par $N(v)$ l'ensemble de tous les sommets voisins de v.

Si $v_i \in N(u)$, alors $v_{i-1} \notin N(v)$.
En effet, si ce n'était pas le cas, $v_1\, v_2\, \ldots\, v_{i-1}\, v_n\, v_{n-1}\, v_{n-2}\, \ldots\, v_{i+1}\, v_i\, v_1$ serait un cycle hamiltonien dans $G^\star$, comme l'illustre la figure ci-dessous.



Ainsi, pour chaque sommet voisin de u, il existe un sommet de $V - \{v\}$ non voisin de v. Mais alors, $\deg(v) \leqslant (n-1) - \deg(u)$ dans $G^\star$: en effet, parmi les $n - 1$ sommets de G distincts de v, au moins $\deg(u)$ d'entre eux ne sont pas voisins de v.

Il en résulte $\deg(u) + \deg(v) \leqslant n - 1$, ce qui vient contredire l'inégalité d'Ore.

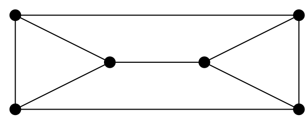
Corollaire : théorème de Dirac (1952)

Soit G *un graphe simple avec* $n \geqslant 3$ *sommets. Si* $\deg(v) \geqslant \frac{n}{2}$ *pour chaque sommet, alors* G *est hamiltonien.*

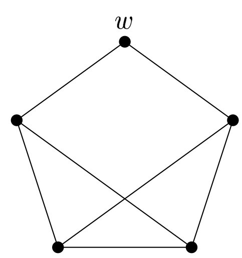
5.4 Démontrer le théorème de Dirac à partir du théorème d'Ore.

Exemples

1) Pour le graphe ci-contre, on a $n = 6$ et $\deg(v) = 3$ pour chaque sommet v. Le théorème de Dirac s'applique donc et le graphe est hamiltonien.



2) Dans le graphe ci-contre, on a $n = 5$ et $\deg(w) = 2$. Le théorème de Dirac ne s'applique donc pas.
Par contre, on a $\deg(u) + \deg(v) \geqslant 5$ pour toute paire de sommets non voisins. Le théorème d'Ore s'applique, de sorte que le graphe est hamiltonien.
Cet exemple montre la plus grande généralité de la condition d'Ore.



Conditions nécessaires pour qu'un graphe soit hamiltonien

Commençons par énoncer les règles fondées sur le fait que tout cycle hamiltonien doit contenir exactement deux arêtes par sommet.

Proposition *Pour qu'un graphe* G *soit hamiltonien, il faut que les règles ci-dessous soient vérifiées.*

Règle 1 *Pour tout sommet v de* G*, on doit avoir* $\deg(v) \geqslant 2$.

Règle 2 *Si un sommet v est de degré* 2*, les deux arêtes passant par v doivent être incluses dans tout cycle hamiltonien.*

Règle 3 *Un cycle hamiltonien ne peut pas contenir de sous-cycle ; autrement dit, au cours de la construction d'un cycle hamiltonien, aucun cycle ne peut être formé avant que tous les sommets aient été visités.*

Règle 4 *Si, au cours de la construction d'un cycle hamiltonien, deux arêtes passant par un sommet v s'imposent, toutes les autres arêtes passant par ce sommet devront être supprimées.*

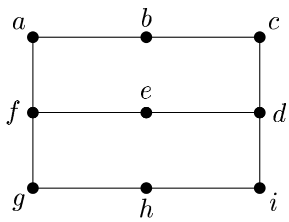
5.5 1) Montrer qu'un cycle hamiltonien est toujours de longueur n, où $n = |\mathrm{V}| =$ nombre de sommets.

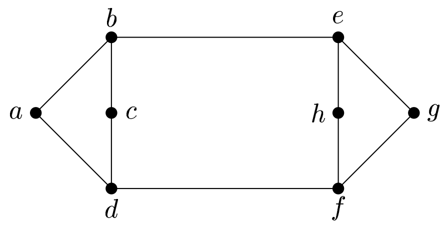
Indication : utiliser le lemme des poignées de mains.

2) Justifier la règle 3 : un cycle hamiltonien ne peut pas contenir de sous-cycle.

Indication : utiliser le résultat de l'exercice 3.5.

5.6 Expliquer pourquoi les graphes suivants ne sont pas hamiltoniens.

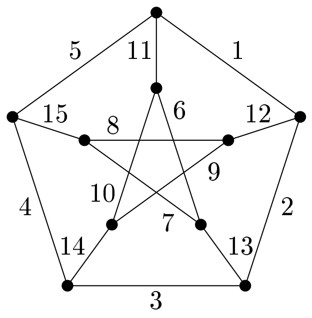
1) 

2) 

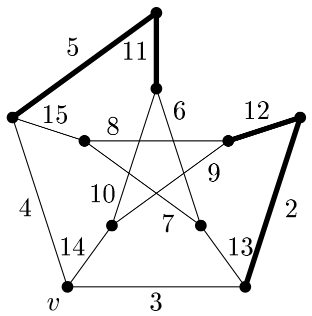
Ces règles, qui sont des conditions nécessaires à l'existence d'un cycle hamiltonien, sont surtout employées quand il s'agit de montrer qu'un graphe n'est pas hamiltonien. La stratégie consiste à essayer de construire un cycle hamiltonien et de montrer qu'à une certaine étape, il est impossible d'aller plus loin.

Exemple Montrons que le graphe de Petersen n'est pas hamiltonien.

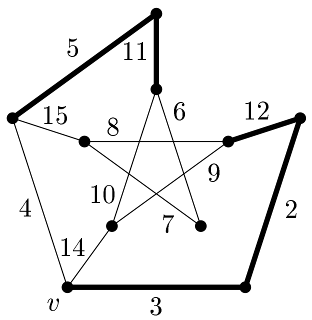
Pour la commodité de la démonstration, on numérote les arêtes plutôt que les sommets.
La stratégie consiste à essayer de construire un cycle hamiltonien en exploitant la symétrie du graphe pour éliminer certaines arêtes. La construction en plusieurs étapes conduira à une contradiction.
Supposons par l'absurde que le graphe soit hamiltonien. À cause de la règle 3, on ne peut pas prendre toutes les arêtes de 1 à 5.



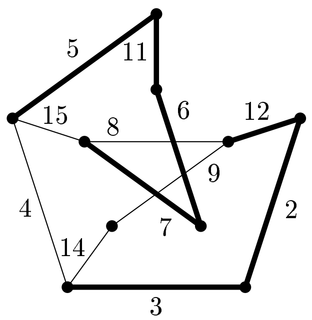
Pour des raisons de symétrie, le choix de celle qui est supprimée est sans importance. Supposons que ce soit l'arête 1. Les arêtes 5 et 11, de même que les arêtes 2 et 12, doivent faire partie du cycle hamiltonien, en vertu de la règle 2.



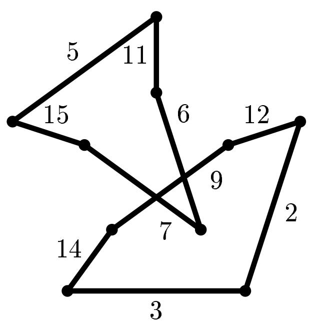
Une des arêtes 3 ou 4 au moins (voire les deux) doit être incluse, sinon le sommet v serait de degré 1, enfreignant la règle 1. Par symétrie, choisissons l'arête 3.
La règle 4 stipule alors que l'on peut supprimer l'arête 13.



Les arêtes 6 et 7 doivent être incluses (règle 2) et l'arête 10 exclue (règle 4).



Les arêtes 9 et 14 doivent être incluses (règle 2) et les arêtes 4 et 8 exclues (règle 4).
Enfin, l'arête 15 doit être comprise (règle 2).
Notre construction d'un cycle hamiltonien débouche finalement en deux sous-cycles, ce qui contredit la règle 3. C'est pourquoi le graphe de Petersen n'est pas hamiltonien.

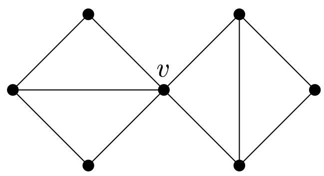


Soit v un sommet d'un graphe G. Considérons le sous-graphe G′ de G obtenu en supprimant le sommet v ainsi que toutes les arêtes de G passant par v.

Un sommet v est appelé un **point d'articulation** si sa suppression « déconnecte » le graphe.

Remarquons qu'il s'agit d'une opération analogue à celle de l'algorithme de Fleury. De la même manière, on s'intéresse aux sommets qui, lorsqu'on les supprime, rendent le graphe G′ non connexe.

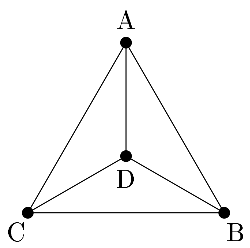
Exemple Dans le graphe ci-dessous, v est un point d'articulation.

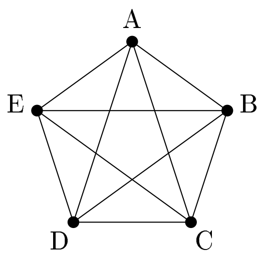


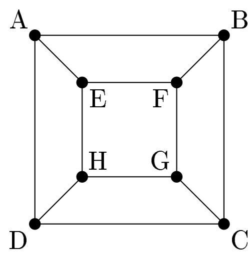
Théorème *Un graphe hamiltonien n'admet aucun point d'articulation.*

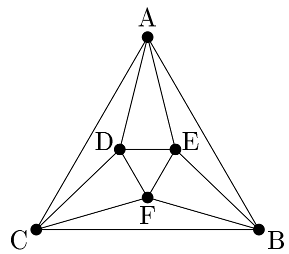
Preuve La suppression d'un sommet au cycle hamiltonien laisse tous les autres sommets sur une même chaîne conservant la connexité de G.

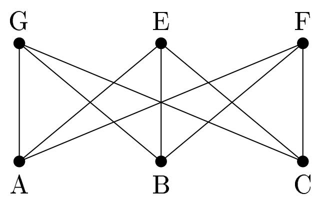
5.7 Parmi les graphes suivants, déterminer ceux qui sont hamiltoniens et trouver, le cas échéant, un cycle hamiltonien.

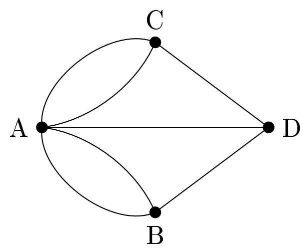
1)


2)


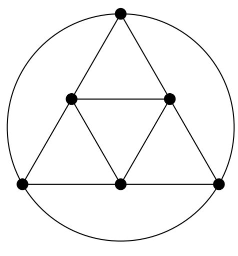
3)


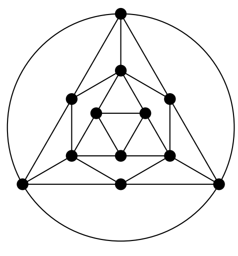
4)


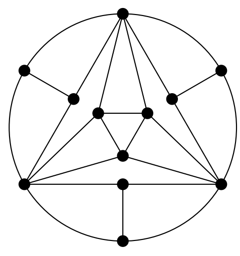
5)


6)


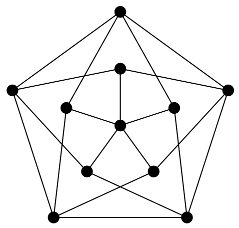
5.8 Déterminer les graphes hamiltoniens :

1)


2)


3)


5.9 Montrer que le graphe de Grötzsch est hamiltonien.

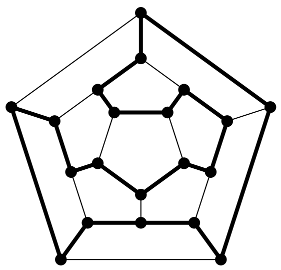


5.10 Donner un exemple comportant au moins six sommets de chacun des graphes suivants :

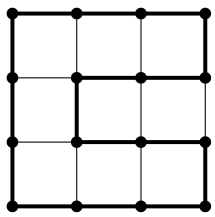
1) un graphe hamiltonien qui n'est pas eulérien ;
2) un graphe eulérien qui n'est pas hamiltonien.

Réponses

5.1



5.2



5.3 1) $\dfrac{n\,(n-1)}{2}$ 2) $\dfrac{(n-1)!}{2}$

5.7
1) hamiltonien : A B C D A
2) hamiltonien : A B C D E A
3) hamiltonien : A E F B C G H D A
4) hamiltonien : A B C D F E A
5) hamiltonien : A E C F B G A
6) hamiltonien : A B D C A

6 Coloriage de graphes

Sans les problèmes de coloriage, la théorie des graphes ne serait pas ce qu'elle est aujourd'hui. La raison en est le célèbre problème des quatre couleurs, déjà mentionné dans l'introduction, qui a stimulé la recherche dans ce domaine au cours du XX[e] siècle.

Dans un graphe, on peut envisager la question du coloriage de deux manières : colorier les sommets ou les arêtes. On se bornera ici aux résultats de base sur le coloriage des sommets.

Coloriage des sommets

On appelle **coloriage des sommets** d'un graphe $\mathrm{G} = (\mathrm{V}\,;\mathrm{E})$ l'opération qui consiste à affecter une couleur à chaque sommet de telle sorte que deux sommets voisins ne portent jamais la même couleur.

Si le coloriage utilise k couleurs, on dit que G est **k-coloriable**.

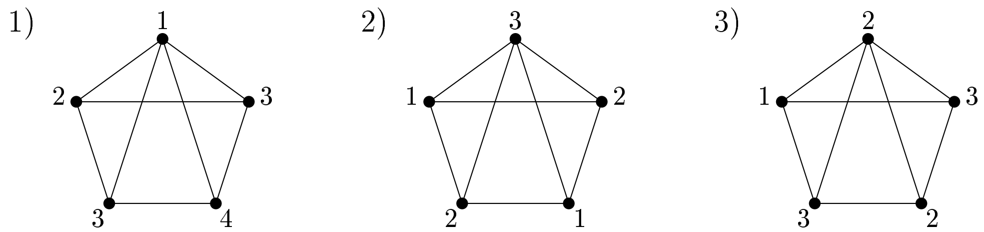
Le **nombre chromatique** de G, noté $\chi(\mathrm{G})$, est le plus petit nombre de couleurs permettant de colorier le graphe G.

Remarques

– Les définitions précédentes ne sont données que pour des graphes simples. Les boucles doivent être exclues, car dans tout k-coloriage les sommets aux extrémités de toutes les arêtes doivent avoir des couleurs différentes. Dans ce cas, un sommet qui comporterait une boucle devrait avoir deux couleurs. On exclut aussi les arêtes multiples entre deux sommets, car celles-ci ne changent rien à la nature du coloriage.

– Pour des raisons de commodité, on décrit les couleurs par des nombres $1, 2, \ldots$ que l'on écrit à côté des sommets concernés.

Exemples Les graphes 1) et 2) illustrent un coloriage de G avec respectivement 4 et 3 couleurs, alors que le graphe 3) n'est pas un coloriage de G.



Le problème le plus important est de calculer $\chi(\mathrm{G})$.

6.1 Trouver le nombre chromatique de K_2, K_3, K_4, K_5 et K_n.

6.2 Trouver le nombre chromatique du graphe cyclique C_n.

Indication : distinguer les cas n pair et n impair.

6.3 Trouver le nombre chromatique d'un arbre.

En général, pour montrer que le nombre chromatique d'un graphe G donné vaut k, il faut vérifier deux choses :

1) trouver un coloriage qui utilise k couleurs ;
2) montrer qu'il n'y a aucun coloriage possible avec moins de k couleurs.

Pour cela, on peut s'aider de la propriété suivante :
si H *est un sous-graphe de* G*, alors* $\chi(\mathrm{H}) \leqslant \chi(\mathrm{G})$.

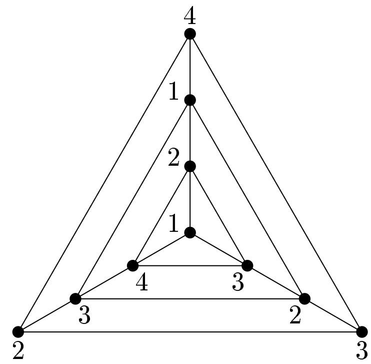
On peut encore remarquer que si G a n sommets, alors $\chi(\mathrm{G}) \leqslant n$.

Exemple
Déterminons le nombre chromatique du graphe G ci-contre.

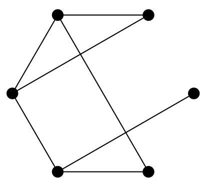
On commence par construire un coloriage avec 4 couleurs ; donc $\chi(\mathrm{G}) \leqslant 4$.

Mais G ne peut pas être colorié avec moins de 4 couleurs, car G contient le graphe complet K_4 ; donc $4 = \chi(\mathrm{K}_4) \leqslant \chi(\mathrm{G})$.

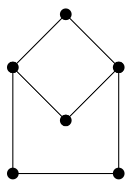
Finalement $\chi(\mathrm{G}) = 4$.

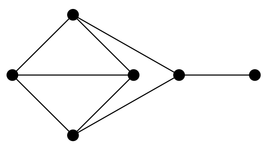


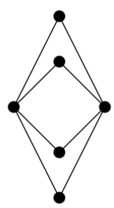
6.4 Déterminer le nombre chromatique du graphe suivant :

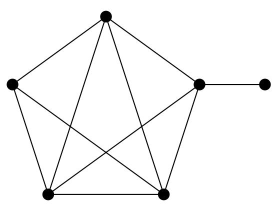


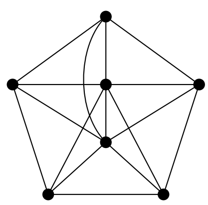
6.5 Déterminer le nombre chromatique pour chacun des graphes suivants :

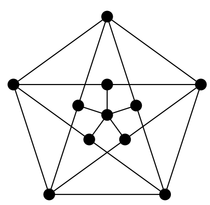
1)


2)


3)


4)


5)


6)


Applications

On peut modéliser la résolution d'un problème de recherche opérationnelle par un coloriage de graphe en groupant dans la même classe des individus ou des objets qui n'entrent pas en conflit.

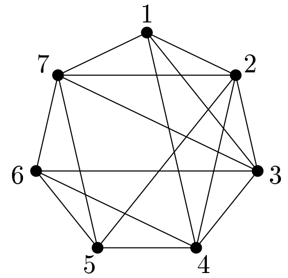
Exemples

Un problème de stockage Supposons qu'une entreprise ait à stocker des produits chimiques. Certains d'entre eux peuvent réagir violemment (explosion, dégagement toxique, etc...) s'ils entrent en contact. Pour cette raison, de tels produits sont dits incompatibles. Pour les garder avec sécurité, il est nécessaire de les conserver dans des zones séparées. Le plus simple serait d'attribuer une zone de stockage par produit, mais on risque alors d'utiliser plus de zones que nécessaires (sauf si tous les produits sont mutuellement incompatibles). Quel est le nombre minimum de zones requises pour conserver tous ces produits de manière sécurisée ?

Ce problème de stockage se traduit en un problème de coloriage d'un graphe. Considérons le graphe $\mathrm{G} = (\mathrm{V}\,;\mathrm{E})$ où V représente l'ensemble des produits chimiques et E l'ensemble d'arêtes reliant deux produits incompatibles. Déterminer le nombre minimum de zones revient à déterminer $\chi(\mathrm{G})$.

Un problème d'horaire Une université doit organiser les horaires des examens. On suppose qu'il y a 7 épreuves à planifier correspondant aux cours numérotés de 1 à 7. Il ne peut y avoir qu'une épreuve par jour. Les paires de cours suivants ont des étudiants communs : 1 et 2, 1 et 3, 1 et 4, 1 et 7, 2 et 3, 2 et 4, 2 et 5, 2 et 7, 3 et 4, 3 et 6, 3 et 7, 4 et 5, 4 et 6, 5 et 6, 5 et 7, 6 et 7. Comment organiser sur une durée minimale ces épreuves de façon qu'aucun étudiant n'ait à passer deux épreuves le même jour ?

À cette fin, construisons le graphe G dont les sommets sont les épreuves numérotées de 1 à 7. Une arête relie deux sommets lorsque les deux cours correspondants possèdent des étudiants communs :



Planifier les examens en un temps minimal consiste à déterminer une k-coloration de G, avec $k = \chi(\mathrm{G})$.

G possède un sous-graphe complet d'ordre 4 (de sommets $1, 2, 3, 4$), donc $4 \leqslant \chi(\mathrm{G})$. Déterminons une partition de G en sous-ensemble stables, à savoir en sous-ensembles ne contenant que des sommets non voisins :

$$S_1 = \{1\,;6\} \qquad S_2 = \{2\} \qquad S_3 = \{3\,;5\} \qquad S_4 = \{4\,;7\}$$

d'où $\chi(G) \leqslant 4$ et finalement $\chi(G) = 4$.

Les examens peuvent être répartis en 4 jours de la manière suivante :
- 1er jour : épreuves des cours 1 et 6 ;
- 2e jour : épreuves du cours 2 ;
- 3e jour : épreuves des cours 3 et 5 ;
- 4e jour : épreuves des cours 4 et 7.

6.6 Dans un congrès, on cherche à planifier l'horaire d'une série de conférences qui doivent être toutes de même durée. Dans le tableau ci-dessous, les étoiles indiquent les interventions qui ne peuvent pas coïncider. Comment procéder pour que la durée totale des interventions soit minimale ?

	a	b	c	d	e	f	g
a	–	⋆	⋆	⋆	–	–	⋆
b	⋆	–	⋆	⋆	⋆	–	⋆
c	⋆	⋆	–	⋆	–	⋆	–
d	⋆	⋆	⋆	–	–	⋆	–
e	–	⋆	–	–	–	–	–
f	–	–	⋆	⋆	–	–	⋆
g	⋆	⋆	–	–	–	⋆	–

6.7 Un gardien de zoo souhaite placer 8 animaux A, B, C, D, E, F, G et H dans des enclos. Le tableau ci-dessous indique par des croix les animaux qui, pour des raisons de sécurité, doivent être placés dans des enclos différents. Déterminer à l'aide d'un graphe convenable le nombre minimum d'enclos qui permet de placer ces animaux de façon judicieuse.

	A	B	C	D	E	F	G	H
A	–	×	–	–	×	×	–	×
B	×	–	×	–	–	×	–	×
C	–	×	–	×	–	×	×	×
D	–	–	×	–	×	×	×	–
E	×	–	–	×	–	×	×	–
F	×	×	×	×	×	–	–	–
G	–	–	×	×	×	–	–	×
H	×	×	×	–	–	–	×	–

Algorithme glouton

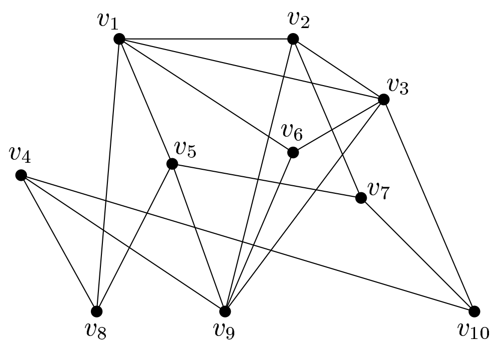
Il y a un algorithme naïf pour décider si un graphe G de n sommets peut être colorié avec k couleurs. Il suffit de vérifier si l'un des k^n coloriages est acceptable. En itérant cet algorithme pour un nombre croissant de k couleurs, on obtient un algorithme pour calculer le nombre chromatique, mais le temps pour l'effectuer croît exponentiellement avec le nombre des sommets.

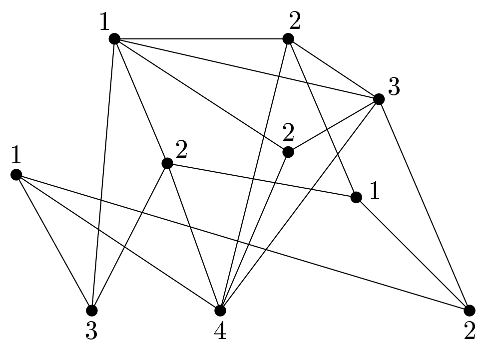
Trouver le nombre chromatique d'un graphe donné est un problème difficile. On ne connaît actuellement aucun algorithme qui fonctionne en temps polynomial et la plupart des spécialistes pensent qu'il n'en existe aucun.

Pourtant, il existe une méthode simple de coloriage : elle consiste à numéroter les sommets du graphe et à colorier successivement chaque sommet avec la première couleur qui n'a pas encore été attribuée à ses voisins. Malheureusement, ce procédé ne fournit pas forcément un coloriage minimum.

On procède comme suit :
- On numérote arbitrairement les sommets de G, à savoir $v_1, v_2, \ldots, v_n$, de même que les couleurs à disposition.
- On affecte la couleur 1 à v_1.
- On considère ensuite le sommet suivant v_2 et on lui attribue la première couleur non déjà attribuée à ses voisins. Dans ce cas, c'est la couleur 1 ou 2.
- Plus généralement, soit le sommet v_i tel que tous les sommets précédents $v_1, v_2, \ldots, v_{i-1}$ soient déjà coloriés. On attribue alors à v_i la première couleur non déjà attribuée à ses voisins.
- On poursuit de même jusqu'à colorier ainsi tous les sommets.

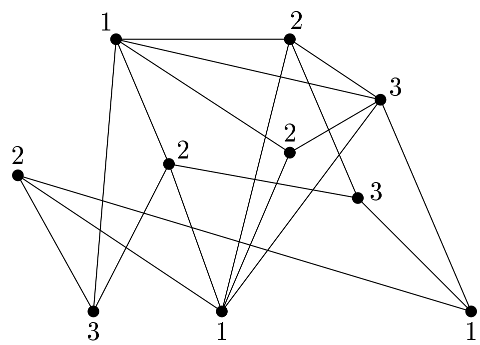
Exemple Voici l'effet de l'algorithme glouton sur le graphe G :





L'application de l'algorithme glouton montre que le graphe G est 4-coloriable.

Pourtant son nombre chromatique $\chi(\mathrm{G}) = 3$, comme le montre la figure ci-contre.

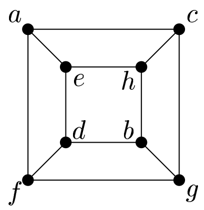


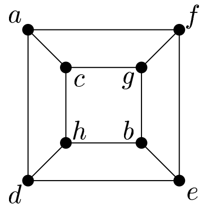
Remarques
- L'efficacité de l'algorithme glouton dépend beaucoup de l'ordre initial donné aux sommets. Il y a $n!$ ordres possibles et, si l'on veut les essayer tous, l'algorithme requiert un temps exceptionnel.

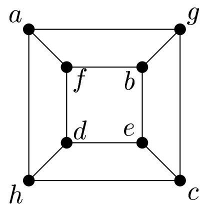
– Il peut arriver que, dans la numérotation des sommets, l'on tombe précisément sur celle qui est associée à un coloriage minimum.

Malgré le gaspillage possible, cet algorithme est utilisé en théorie et en pratique.

6.8 Utiliser l'algorithme glouton pour colorier les sommets du graphe suivant en respectant dans chaque cas l'ordre proposé.

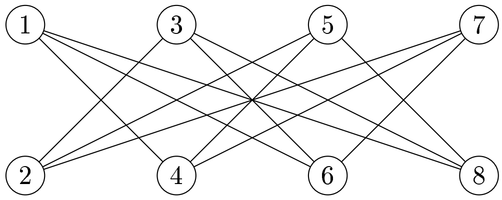
1) 

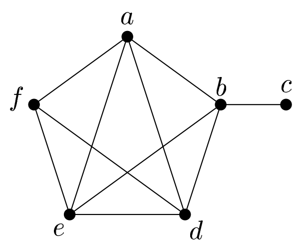
2) 

3) 

Quelle est la valeur de $\chi(\mathrm{G})$?

6.9 Utiliser l'algorithme glouton pour colorier les graphes suivants :

1) 

2) 

Théorème *Si* G *est un graphe simple tel que le degré maximum des sommets soit d, alors* $\chi(\mathrm{G}) \leqslant d + 1$.

Preuve Quelle que soit la numérotation des sommets, l'algorithme glouton n'utilise jamais plus de $d + 1$ couleurs, puisqu'un sommet n'a jamais plus de d voisins.

6.10 Montrer qu'ajouter une arête à un graphe augmente son nombre chromatique d'au plus 1.

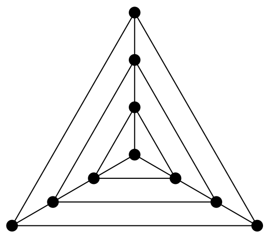
Avec plus d'efforts, on peut améliorer le résultat du théorème précédent.

Théorème de Brooks (1941) *Si* G *est un graphe simple et connexe sans être un graphe complet, et si le plus haut degré des sommets de* G *est d ($d \geqslant 3$), alors* $\chi(\mathrm{G}) \leqslant d$.

Nous ne démontrerons pas ce théorème, mais nous allons en illustrer l'emploi.

Exemple

Considérons à nouveau le graphe G ci-contre.

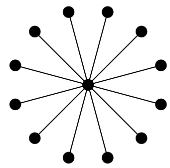


Nous avons déjà vu que $4 \leqslant \chi(\mathrm{G})$, du fait que G contient le graphe complet K_4.

Par ailleurs, G satisfait les conditions du théorème de Brooks avec $d = 4$, d'où $\chi(\mathrm{G}) \leqslant 4$.

On conclut que $\chi(\mathrm{G}) = 4$.

Malheureusement, la situation n'est pas toujours aussi favorable. En particulier, si G contient un petit nombre de sommets de degré élevé, la borne proposée par le théorème de Brooks n'est du tout satisfaisante.



Par exemple, considérons le graphe ci-contre. D'après le théorème de Brooks, $\chi(\mathrm{G}) \leqslant 12$, alors que $\chi(\mathrm{G}) = 2$.

6.11 Dessiner deux graphes non isomorphes, simples et connexes, comportant 5 sommets, dont le plus haut degré est d, et tels que $\chi(\mathrm{G}) = d + 1$.

Polynôme chromatique

Malgré les résultats obtenus ci-dessus, déterminer de manière efficace le nombre chromatique reste encore un problème non résolu. Nous avons vu que la méthode consistant à essayer toutes les possibilités devient rapidement impraticable. Pourtant, il existe des algorithmes qui améliorent substantiellement la recherche du nombre chromatique. Nous allons en présenter un qui utilise des techniques algébriques.

Soit G un graphe simple. Notons $\mathrm{P_G}(\lambda)$ le nombre de façons de colorier les sommets de G avec λ couleurs. La fonction $\mathrm{P_G}(\lambda)$ s'appelle le **polynôme chromatique**[1] de G.

6.12 Déterminer le polynôme $\mathrm{P_G}(\lambda)$ si G est le graphe nul (sans arêtes) à n sommets.

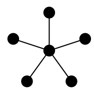
6.13 Déterminer le polynôme $\mathrm{P_G}(\lambda)$ si G est le graphe complet K_n.

Proposition *Si* G *est un arbre à n sommets, alors* $\mathrm{P_G}(\lambda) = \lambda\,(\lambda - 1)^{n-1}$.

1. Il n'est pas évident a priori que le nombre de λ-coloriages d'un graphe G soit un polynôme en λ. Ce résultat sera établi plus tard (théorème de Birkhoff).

Preuve On montre le résultat par récurrence sur le nombre de sommets n.
Si $n = 1$, on a évidemment $\mathrm{P_G}(\lambda) = \lambda$.
Si $n > 1$, l'arbre possède une extrémité, c'est-à-dire un sommet a de degré 1. Ôtons de G le sommet a et l'arête issue de a. Le graphe H restant est un arbre à $n - 1$ sommets. L'hypothèse de récurrence implique $\mathrm{P_H}(\lambda) = \lambda\,(\lambda - 1)^{n-2}$.
Dans un coloriage de G, le sommet a peut recevoir l'une quelconque des λ couleurs, à part celle de son unique voisin, d'où $\mathrm{P_G}(\lambda) = \mathrm{P_H}(\lambda)\,(\lambda - 1)$.
On conclut finalement que $\mathrm{P_G}(\lambda) = \lambda\,(\lambda - 1)^{n-2}\,(\lambda - 1) = \lambda\,(\lambda - 1)^{n-1}$.

Il est clair que :
– si $\lambda < \chi(\mathrm{G})$, alors $\mathrm{P_G}(\lambda) = 0$;
– si $\lambda \geqslant \chi(\mathrm{G})$, alors $\mathrm{P_G}(\lambda) > 0$.
Le nombre chromatique de G est ainsi le plus petit entier positif λ pour lequel $\mathrm{P_G}(\lambda) > 0$. C'est pourquoi trouver une méthode pour calculer $\mathrm{P_G}(\lambda)$, c'est trouver une méthode pour calculer $\chi(\mathrm{G})$.

6.14 Écrire les polynômes chromatiques des graphes suivants :
1) le graphe complet K_6 ;
2) le graphe bipartite complet $\mathrm{K}_{1,5}$: 

De combien de manières ces graphes peuvent-ils être coloriés avec 7 couleurs ?

Le théorème suivant va fournir une méthode systématique pour obtenir le polynôme chromatique d'un graphe à partir du polynôme chromatique d'un graphe nul.

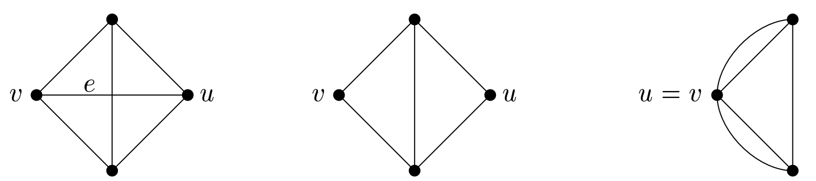
Mais, pour pouvoir l'énoncer, nous devons d'abord définir deux opérations sur les graphes.

Soit une arête e reliant des sommets u et v d'un graphe G.

Le graphe $\mathrm{G} - e$ est le graphe obtenu à partir de G en supprimant l'arête e.

Le graphe $\mathrm{G} \setminus e$ est le graphe obtenu en contractant e, c'est-à-dire en supprimant e et en identifiant les sommets u et v.

Exemple Nous avons représenté ci-dessous les graphes G, $\mathrm{G} - e$ et $\mathrm{G} \setminus e$:



Théorème (suppression-contraction) *Soient un graphe simple* G, $\mathrm{G} - e$ *le graphe obtenu en supprimant une arête* e *et* $\mathrm{G} \setminus e$ *le graphe obtenu en contractant cette arête* e. *Alors :*

$$\mathrm{P_G}(\lambda) = \mathrm{P}_{\mathrm{G}-e}(\lambda) - \mathrm{P}_{\mathrm{G}\setminus e}(\lambda)$$

Preuve Les coloriages de $G - e$ peuvent se partager en deux classes disjointes : la classe C_1 où u et v sont de couleurs différentes et la classe C_2 où u et v sont de même couleur. Posons $N_1 = |C_1|$ et $N_2 = |C_2|$.

1^er^ cas : u et v sont de couleurs différentes.

La suppression de l'arête e dans G n'affecte en rien le coloriage de G, d'où $N_1 = P_G(\lambda)$.

2^nd^ cas : u et v sont de même couleur.

Le nombre de coloriages de $G - e$ vaut dans ce cas $N_2 = P_{G \smallsetminus e}(\lambda)$.

Il en résulte $P_{G-e}(\lambda) = N_1 + N_2 = P_G(\lambda) + P_{G \smallsetminus e}(\lambda)$, d'où le théorème.

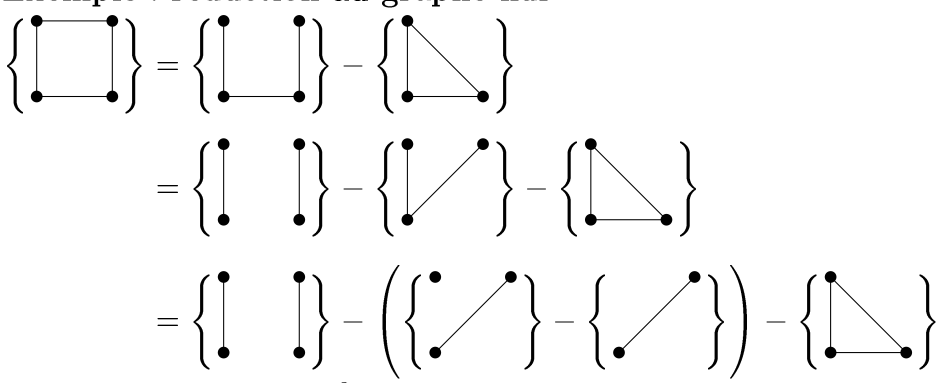
L'intérêt du théorème ci-dessus est de donner une formule de récurrence pour calculer $P_G(\lambda)$ selon l'une des méthodes suivantes :

1) après avoir retiré successivement toutes les arêtes, on parvient au graphe nul avec n sommets ;
2) dans l'autre sens, en ajoutant chaque fois une arête, on parvient au graphe complet K_n.

Pour n fixé, quand le nombre d'arêtes est petit, la première méthode est préférable ; quand le nombre d'arêtes est grand, c'est la seconde méthode qui est préférable.

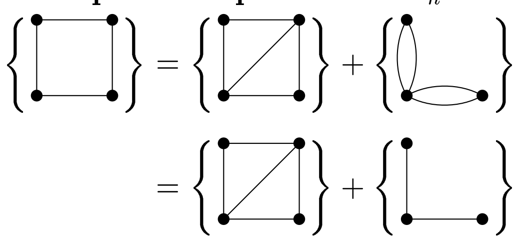
Illustrons ces deux méthodes pour le graphe cyclique C_4, en symbolisant le polynôme chromatique d'un graphe par le graphe lui-même, dessiné entre accolades.

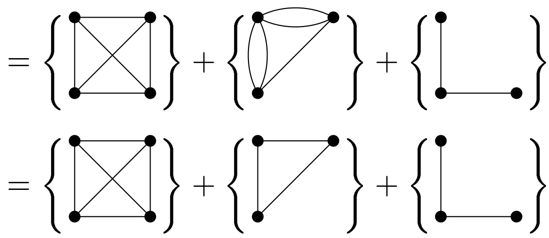
Exemple : réduction au graphe nul



$$= \big(\lambda\,(\lambda-1)\big)^2 - \big(\lambda^2\,(\lambda-1) - \lambda\,(\lambda-1)\big) - \lambda\,(\lambda-1)\,(\lambda-2)$$
$$= \lambda^4 - 4\,\lambda^3 + 6\,\lambda^2 - 3\,\lambda$$

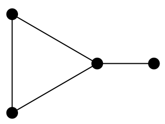
Exemple : complétion en K_n

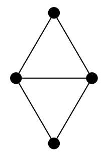




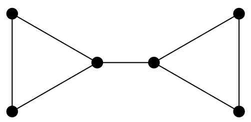
$$= \lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda-3) + \lambda\,(\lambda-1)\,(\lambda-2) + \lambda\,(\lambda-1)^2$$
$$= \lambda^4 - 4\,\lambda^3 + 6\,\lambda^2 - 3\,\lambda$$

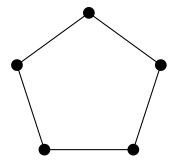
6.15 Déterminer le polynôme chromatique des graphes suivants :

1) 

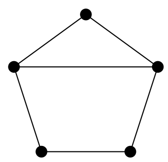
2) 

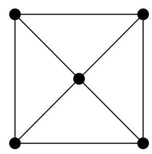
6.16 Déterminer le polynôme chromatique des graphes suivants :

1) 

2) 

6.17 Déterminer le polynôme chromatique des graphes suivants :

1) 

2) 

6.18 Construire des graphes possédant les polynômes chromatiques suivants :

1) $\lambda^2\,(\lambda-1)\,(\lambda-2)$

2) $\lambda\,(\lambda-1)^2\,(\lambda-2)$

3) $\lambda\,(\lambda-1)\,(\lambda-2)^2$

4) $\lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda-3)$

6.19 Trouver un graphe admettant $\lambda^5 - 6\,\lambda^4 + 11\,\lambda^3 - 6\,\lambda^2$ pour polynôme chromatique.

Théorème de Birkhoff *Soit* G *un graphe simple avec* n *sommets et* m *arêtes. Alors* $\mathrm{P_G}(\lambda)$ *est un polynôme unitaire de degré* n *en* λ*, à coefficients entiers et de terme constant nul. De plus, ses coefficients alternent en signe et le coefficient de* λ^{n-1} *vaut* $-m$*.*

Preuve La démonstration se fait par récurrence sur le nombre d'arêtes du graphe dont le nombre de sommets n est fixé.

Si $m = 0$, G est le graphe nul avec n sommets, donc $\mathrm{P_G}(\lambda) = \lambda^n$.

Soit G un graphe d'ordre n avec m arêtes et soit e une arête quelconque de G. À la fois $\mathrm{G} - e$ et $\mathrm{G} \smallsetminus e$ (après suppression des arêtes multiples si nécessaires) sont des graphes simples avec au plus $m - 1$ arêtes. Ainsi, par hypothèse de récurrence, on a :

$$\begin{aligned} \mathrm{P}_{\mathrm{G}-e}(\lambda) &= \lambda^n - a_{n-1}\,\lambda^{n-1} + \ldots + (-1)^{n-1}\,a_1\,\lambda \\ \mathrm{P}_{\mathrm{G}\smallsetminus e}(\lambda) &= \lambda^{n-1} - b_{n-2}\,\lambda^{n-2} + \ldots + (-1)^{n-2}\,b_1\,\lambda \end{aligned}$$

où $a_1, \ldots, a_{n-1}, b_1, \ldots, b_{n-2}$ sont des entiers non négatifs et $a_{n-1} = m - 1 =$ nombre d'arêtes de $\mathrm{G} - e$.

D'après le théorème de suppression-contraction, $\mathrm{P_G}(\lambda) = \mathrm{P}_{\mathrm{G}-e}(\lambda) - \mathrm{P}_{\mathrm{G}\smallsetminus e}(\lambda)$, de sorte que :

$$\mathrm{P_G}(\lambda) = \lambda^n - (a_{n-1} + 1)\,\lambda^{n-1} + \ldots + (-1)^{n-1}\,(a_1 + b_1)\,\lambda$$

Puisque $a_{n-1} + 1 = m$, $\mathrm{P_G}(\lambda)$ vérifie toutes les propriétés annoncées.

6.20 Montrer que les polynômes suivants ne sont pas des polynômes chromatiques de graphes.

1) $\lambda^7 - \lambda^6 + 1$ 2) $\lambda^4 - 3\,\lambda^3 + \lambda^2 - \lambda$

Réponses

6.1 $\chi(\mathrm{K}_n) = n$

6.2 $\chi(\mathrm{C}_n) = \begin{cases} 2 & \text{si } n \text{ est pair} \\ 3 & \text{si } n \text{ est impair} \end{cases}$

6.3 2 (s'il possède au moins 2 sommets)

6.4 3

6.5

1) 3 2) 3 3) 2

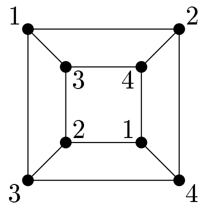
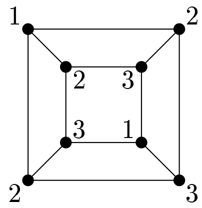
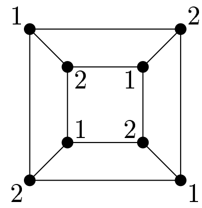
4) 4 5) 5 6) 4

6.6

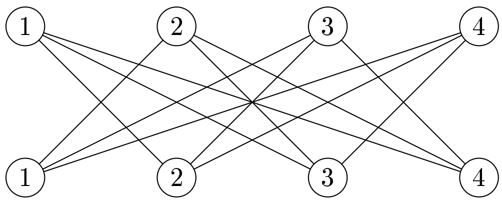
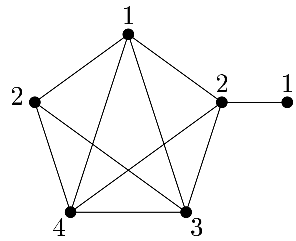
a, e, f	b	c, g	d

6.7 4 enclos sont nécessaires et suffisent :

A C	B E	D H	F G

6.8 1)  2)  3) 

$\chi(\mathrm{G}) = 2$

6.9 1)  2) 

6.11 K_5 et C_5

6.12 $\mathrm{P}_{\mathrm{G}}(\lambda) = \lambda^n$

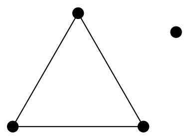
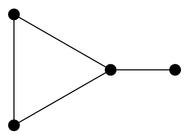
6.13 $\mathrm{P}_{\mathrm{G}}(\lambda) = \lambda\,(\lambda-1)\,(\lambda-2)\,\cdots\,(\lambda-n+1)$

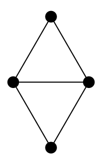
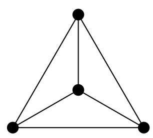
6.14 1) $\lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda-3)\,(\lambda-4)\,(\lambda-5)$ 5040 2) $\lambda\,(\lambda-1)^5$ 54 432

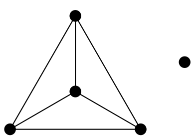
6.15 1) $\lambda\,(\lambda-1)^2\,(\lambda-2)$ 2) $\lambda\,(\lambda-1)\,(\lambda-2)^2$

6.16 1) $\lambda\,(\lambda-1)^3\,(\lambda-2)^2$ 2) $\lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda^2-2\,\lambda+2)$

6.17 1) $\lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda^2-3\,\lambda+3)$ 2) $\lambda\,(\lambda-1)\,(\lambda-2)\,(\lambda^2-5\,\lambda+7)$

6.18 1)  2) 

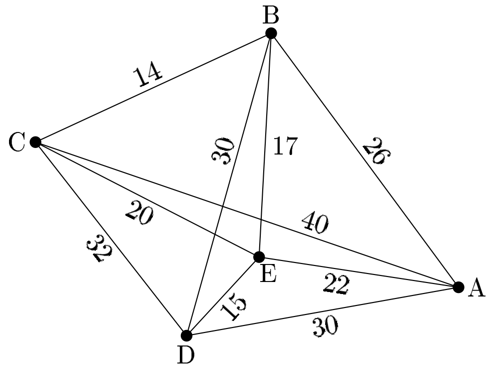
3)  4) 

6.19 

7 Algorithmes efficaces

En théorie des graphes, nous avons déjà vu ce qu'est un algorithme, à savoir la donnée d'une série d'instructions qui permettent, pas à pas, d'obtenir la solution d'un problème donné.

Dans la pratique, la découverte d'un algorithme pour résoudre un problème n'est pas suffisante. Il faut encore se poser la question de son efficacité. Pour mieux comprendre ce que cela veut dire, revenons au problème du voyageur de commerce et supposons que celui-ci, habitant la ville A, dispose de la carte routière suivante avec les distances entre les villes (il doit revenir en A).



7.1 La manière la plus évidente de résoudre ce problème est d'employer la « force brute » :

1) faire la liste de tous les itinéraires possibles ;
2) calculer la longueur de chacun d'entre eux ;
3) sélectionner celui qui est le plus court.

7.2

1) Combien y a-t-il d'itinéraires possibles dans l'exercice précédent ?
2) En supposant que le graphe soit complet, combien y a-t-il d'itinéraires possibles pour
   (a) 6 villes ?
   (b) 7 villes ?
   (c) n villes ?
3) Imaginons qu'un ordinateur très rapide puisse décrire un itinéraire et calculer sa longueur en 10 microsecondes. Combien de temps lui faudrait-il, avec 20 villes, pour déterminer le trajet le plus court en utilisant la méthode de la « force brute » ?

La technique de la « force brute » ou autrement dit de l'énumération exhaustive peut être très efficace sur de petits exemples, mais quand les données sont d'une certaine importance, elle devient complètement impraticable. À part le fait d'exiger qu'un algorithme ait chacune de ses étapes bien définies et fournisse la solution en un nombre fini d'étapes, il faut donc encore ajouter des considérations d'ordre pratique. Est-ce que le problème, pour des données de taille modérée, peut être résolu par tel ou tel algorithme en un temps raisonnable ? Répondre à cette question implique de dénombrer chacune des opérations qui sont faites à chaque étape.

Par exemple, dans l'algorithme de Fleury, on peut vérifier que ce nombre est proportionnel à n où n est le nombre d'arêtes. Pour l'algorithme de connexion minimale, ce nombre est à peu près proportionnel à n^2 où n est le nombre de sommets.

Dans ces cas ou dans d'autres où le temps utilisé pour faire fonctionner l'algorithme est un polynôme en n, où n représente un certain paramètre associé à la quantité des données implémentées dans l'ordinateur, on parle d'algorithme **en temps polynomial**.

Il existe des algorithmes dont le temps de fonctionnement est de l'ordre d'une puissance de n. C'est le cas par exemple lorsqu'il faut examiner toutes les parties d'un ensemble à n éléments, puisqu'un tel ensemble a 2^n parties. On dit alors que ces algorithmes sont **en temps exponentiel**; on les appelle aussi algorithmes **gloutons**.

7.3 Imaginons que l'on utilise un ordinateur capable d'effectuer 1000 opérations à la seconde. Compléter le tableau suivant qui donne les temps approximatifs nécessaires pour effectuer divers algorithmes, les uns en temps polynomial (d'ordre n et n^3), les autres en temps exponentiel (d'ordre 2^n et 3^n).

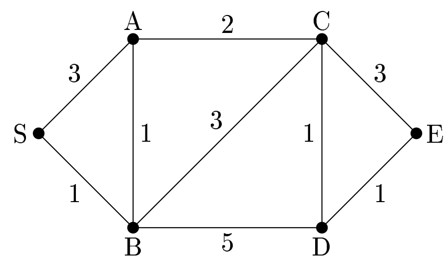
	$n = 10$	$n = 50$	$n = 100$
n			
n^3			
2^n			
3^n			

L'exercice 7.3 montre pourquoi l'on considère les algorithmes en temps polynomial comme efficaces, alors que les algorithmes en temps exponentiel sont considérés de peu d'utilité sauf dans les exemples de petite taille.

Algorithme de Dijkstra

L'algorithme de Dijkstra est un algorithme en temps polynomial (d'ordre n^2) qui permet de trouver quelle est la longueur du plus court chemin qui joint deux villes déterminées lorsqu'on connaît la longueur des routes.

Par exemple, considérons la situation suivante où les sommets représentent des villes, les arêtes des routes et les nombres des distances.



On cherche à déterminer le plus court chemin allant de S à E.

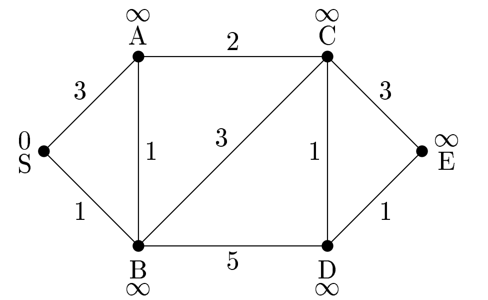
L'idée générale de l'algorithme de Dijkstra est de parcourir le graphe en allant de S à E en attribuant de proche en proche à chaque sommet un poids qui est égal à la plus petite des distances entre S et ce sommet.

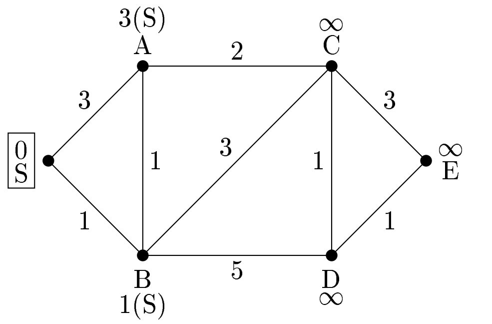
Plus précisément, on procède comme suit.

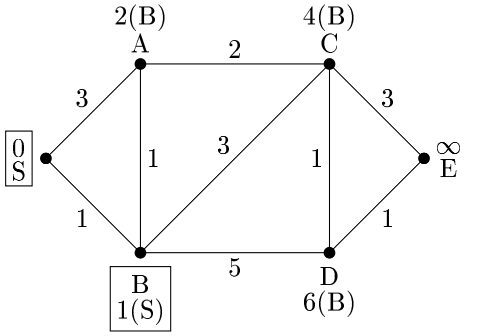
Initialisation Tous les sommets sont non marqués et ont un poids provisoire infini, sauf le sommet de départ qui a un poids nul.

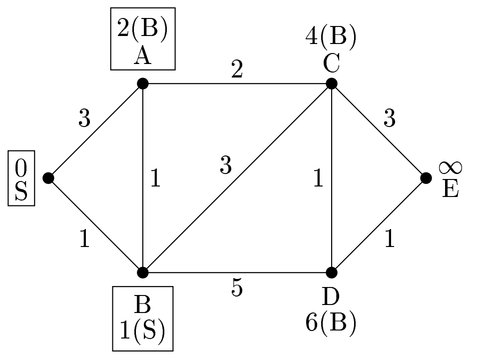
Itérations Tant qu'il existe un sommet non marqué :
- choisir le sommet T qui possède le plus petit poids provisoire ;
- fixer définitivement le poids de T et marquer le sommet T ;
- pour chaque sommet U non marqué et voisin de T :
  - calculer la somme s du poids de T et du poids de l'arête reliant T à U ;
  - si s est inférieur au poids provisoire de U, affecter s à U comme nouveau poids provisoire et le noter s(T) pour indiquer ainsi la provenance de cette dernière affectation, sinon conserver le poids provisoire.

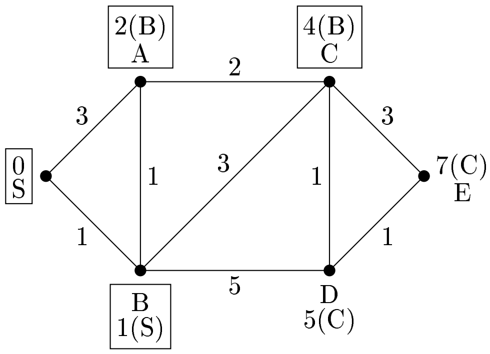
Voici l'application de cet algorithme au graphe précédent :

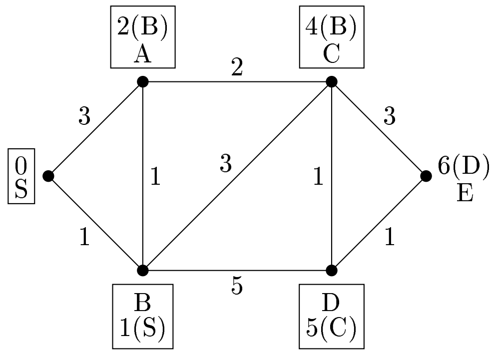


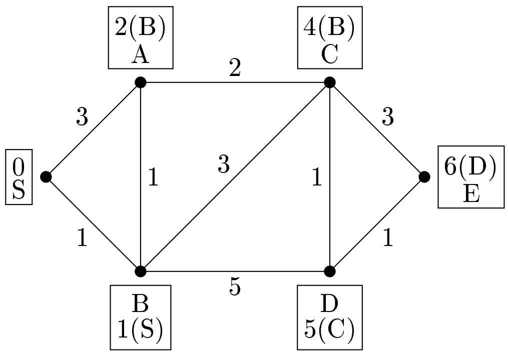










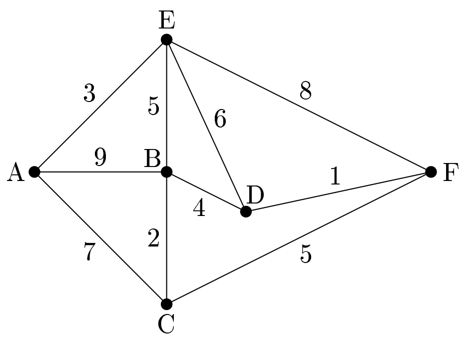


Le plus court chemin a un poids 6.
Il se lit à l'envers : EDCBS.

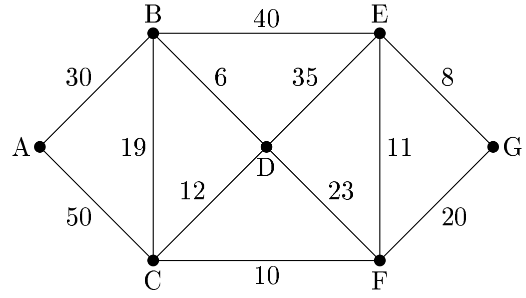
On peut aussi représenter les différentes étapes de l'algorithme exécutées sur cet exemple par un tableau :

S	A	B	C	D	E	sommets marqués
0						
0	3(S)	1(S)				S
	2(B)	1(S)	4(B)	6(B)		S, B
	2(B)		4(B)	6(B)		S, B, A
			4(B)	5(C)	7(C)	S, B, A, C
				5(C)	6(D)	S, B, A, C, D
					6(D)	S, B, A, C, D, E

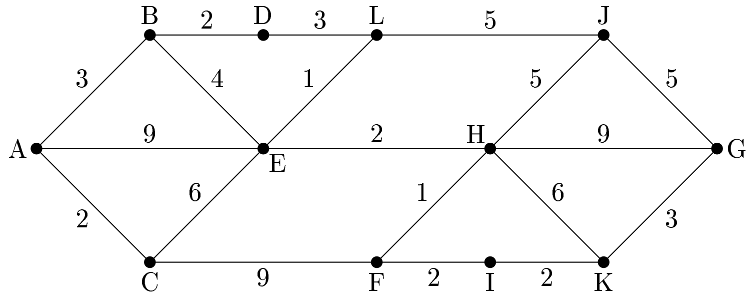
7.4 Utiliser l'algorithme de Dijkstra pour trouver le chemin de poids minimal entre A et F :



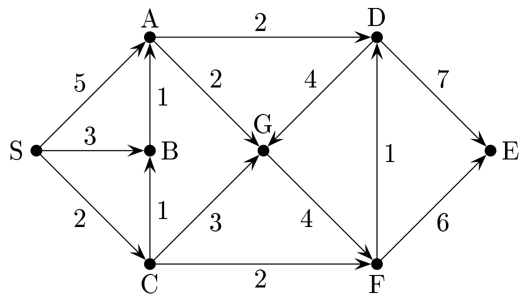
7.5 Utiliser l'algorithme de Dijkstra pour trouver le chemin de poids minimal entre A et G :



7.6 Utiliser l'algorithme de Dijkstra pour trouver le chemin de poids minimal entre A et G :



7.7 Utiliser l'algorithme de Dijkstra pour trouver le chemin de poids minimal entre S et E, en prenant garde au fait qu'il s'agit d'un graphe orienté.



Le problème du postier

Un postier cherche à distribuer ses lettres en parcourant la plus petite distance tout en retournant à son point de départ. Il doit passer évidemment par chaque rue au moins une fois en évitant autant que possible de repasser par un tronçon déjà parcouru.

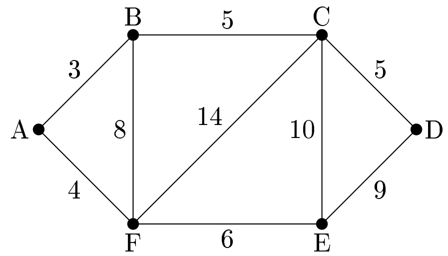
Il y a quelques années, la ville de Zürich commandita une grande étude pour déterminer rationnellement le plan de déneigement de ses rues. C'est, en plus compliqué, un problème semblable à celui du postier. En effet, il est nécessaire en plus de partager habilement la ville en secteurs de manière à occuper chaque véhicule de déneigement.

Le problème peut être reformulé en termes de graphe pondéré où le graphe correspond au réseau des rues et le poids de chaque arête à la longueur de la rue correspondante. Dans cette nouvelle formulation, l'exigence est de trouver un chemin fermé de poids minimal qui inclut chaque arête au moins une fois.

Si le graphe est eulérien, chaque chemin eulérien sera un itinéraire acceptable. Un tel chemin peut être obtenu si nécessaire à l'aide de l'algorithme de Fleury.

Par contre, si le graphe n'est pas eulérien, le problème est beaucoup plus difficile. On connaît cependant un algorithme efficace pour trouver la solution, même s'il est trop compliqué pour être donné ici. La solution générale s'inspire du cas particulier où le graphe est semi-eulérien.

Considérons par exemple le graphe suivant :

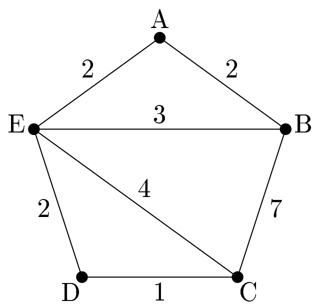


Puisque B et E sont les deux seuls sommets de degré impair, nous pouvons trouver un chemin semi-eulérien allant de B à E en parcourant chaque arête une seule fois.

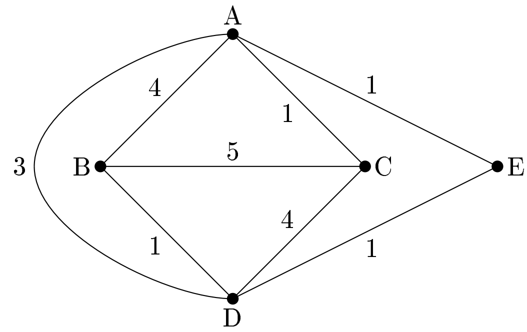
Pour retourner au point de départ en parcourant la plus petite distance possible, nous pouvons trouver le plus court chemin allant de B à E en utilisant l'algorithme de Dijkstra.

La solution du problème du postier est obtenue dans ce cas en joignant le plus court chemin EFAB au chemin semi-eulérien initial BAFBCFECDE. La distance totale vaut donc $13 + 64 = 77$.

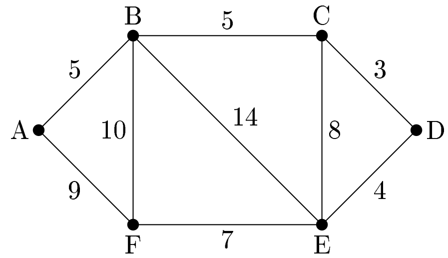
7.8 Résoudre le problème du postier pour le graphe pondéré ci-dessous :



7.9 Résoudre le problème du postier pour le graphe pondéré ci-dessous :



7.10 Résoudre le problème du postier pour le graphe pondéré ci-dessous :



Réponses

7.1

ABCDEA	109	ACBDEA	121	ADBCEA	116	AEBCDA	115
ABCEDA	105	ACBEDA	116	ADBECA	137	AEBDCA	141
ABDECA	131	ACDBEA	141	ADCBEA	115	AECBDA	116
ABDCEA	130	ACDEBA	130	ADCEBA	125	AECDBA	130
ABEDCA	130	ACEBDA	137	ADEBCA	116	AEDBCA	121
ABECDA	125	ACEDBA	131	ADECBA	105	AEDCBA	109

7.2 1) $4! = 24$ 2) $5! = 120$ $6! = 720$ $(n-1)!$ 3) 38 573 années

7.3

	$n = 10$	$n = 50$	$n = 100$
n	0,01 s	0,05 s	0,1 s
n^3	1 s	2 min	16 min
2^n	1 s	35 702 ans	$4 \cdot 10^{19}$ ans
3^n	1 min	$2{,}27 \cdot 10^{13}$ ans	$1{,}63 \cdot 10^{37}$ ans

7.4 AEDF de poids 10

7.5 ABDCFEG de poids 77

7.6 ABEHFIKG de poids 17

7.7 SCFE de poids 10

7.8 BAEDCBECDEB de poids 27

7.9 BACDAEDBCAEDB de poids 24

7.10 CDEFABCEBFEDC de poids 79