3 Chemins & Arbres

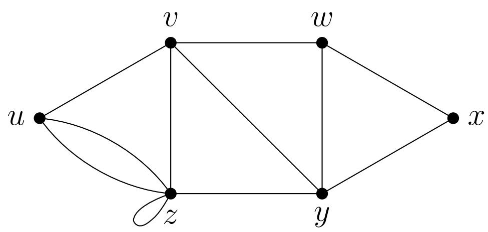
Chemins

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.

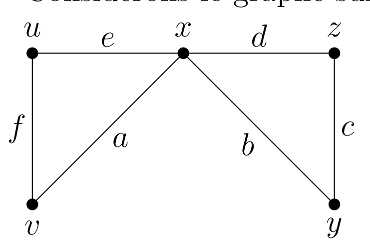
Si $a = b$, on parle d'une chaîne **fermée**, sinon d'une chaîne **ouverte**.

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.

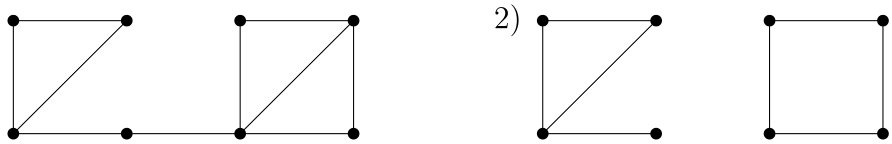
Il est facile de montrer que tout chemin allant d'un sommet à un autre peut être « simplifié » en un chemin simple. Pour cela, il suffit de supprimer les détours.

Connexité

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

3.2 Les graphes suivants sont-ils connexes ?

1) 2)



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.

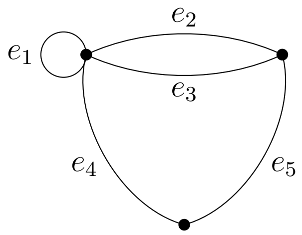
1) Supposons qu'il existe un sommet de degré 1.
   Soit G' le sous-graphe de G obtenu par suppression d'un sommet de degré 1 et de l'arête adjacente à ce sommet. Alors G' est un graphe connexe avec $n-1$ sommets. Vu l'hypothèse de récurrence, il possède au moins $(n-1)-1$ arêtes. Il en résulte que G, 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 $2\,|\mathrm{E}| = \sum\limits_{x\in\mathrm{V}} \deg(x) \geqslant 2\,n$, 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 $x_1, x_2, \ldots, x_k$ 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 $x_k x_j x_{j+1} \ldots x_{k-1} x_k$ 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.

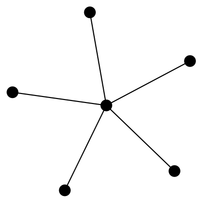
Arbres

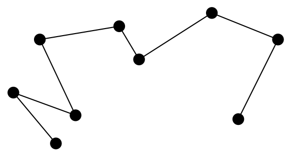
Un **arbre** est un graphe connexe acyclique.

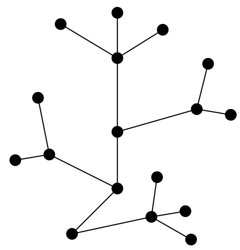
3.4 Quel est le nombre d'arêtes d'un arbre à n sommets ?

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

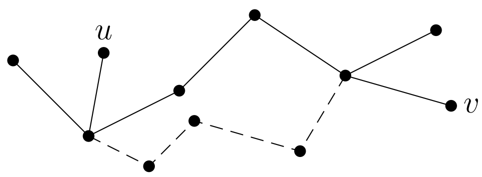
La connexité d'un graphe implique que deux sommets quelconques sont toujours reliés par *au moins un* chemin.

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 cycle, 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.

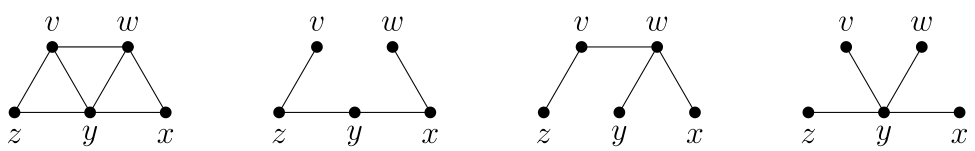
3.6 Prouver qu'un graphe de n sommets et $n-1$ arêtes qui a au moins un cycle a plus d'une composante connexe.

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.

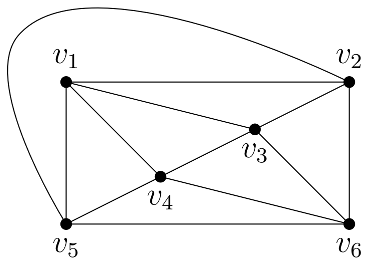
Considérons l'ensemble Ψ de tous les sous-graphes connexes de G contenant tous les sommets de G. L'ensemble Ψ est non vide, vu que $G \in \Psi$.

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

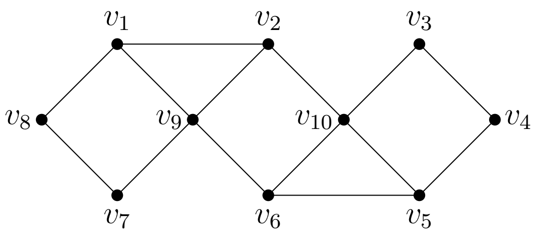
T est acyclique : sinon, T contiendrait un cycle et la suppression d'une arête quelconque de ce cycle donnerait un sous-graphe appartenant à Ψ ayant une arête de moins que T, ce qui contredirait le choix de T.

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

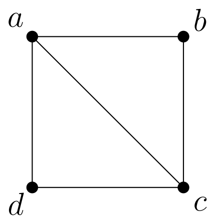
3.7 Trouver un arbre de recouvrement du graphe suivant :



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**.

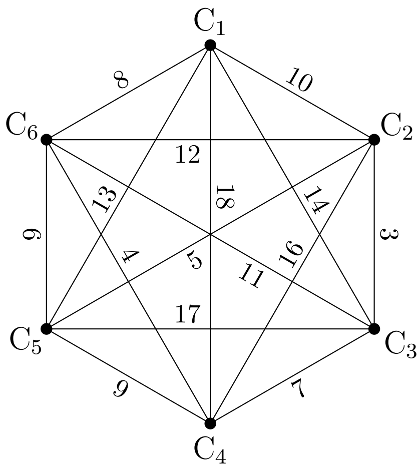
Le problème consiste alors à trouver un arbre de recouvrement de G de poids minimum.

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

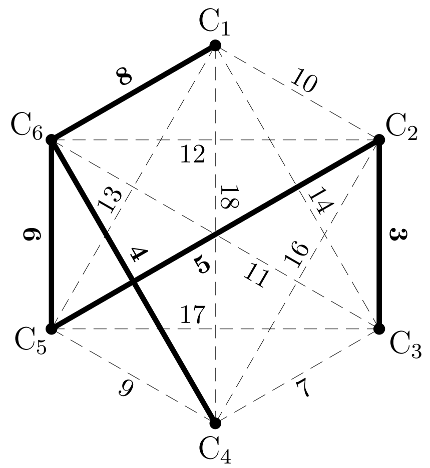
1) Trier les arêtes de E par ordre croissant de poids : $e_1, \ldots, e_{|\mathrm{E}|}$.
2) Poser $\mathrm{F} = \varnothing$.
3) Pour tout i allant de 1 à n, ajouter l'arête e_i à F, pour autant que le graphe $(\mathrm{V}\,;\mathrm{F})$ qui en résulte demeure acyclique.

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 $\mathrm{F} = \{3\,;4\,;5\,;6\,;8\}$.



En effet, l'ajout de l'arête de poids 7 créerait le cycle $\mathrm{C}_2\,\mathrm{C}_3\,\mathrm{C}_4\,\mathrm{C}_6\,\mathrm{C}_5\,\mathrm{C}_2$. 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.

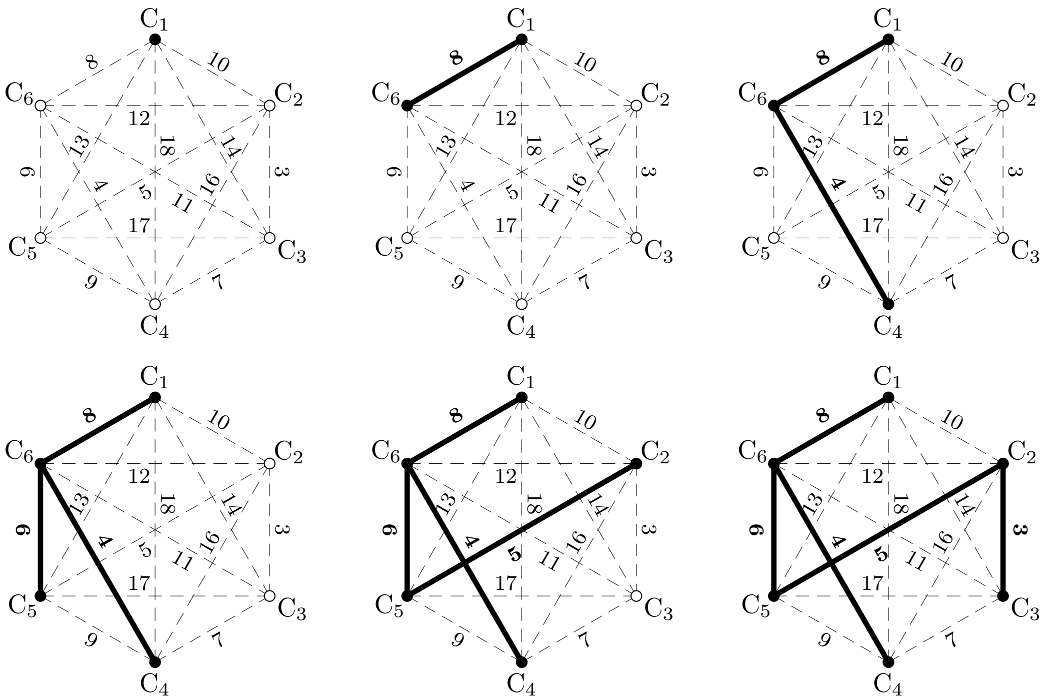
L'algorithme de Prim, que l'on doit à l'informaticien Robert C. Prim en 1957, permet de surmonter ces difficultés.

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

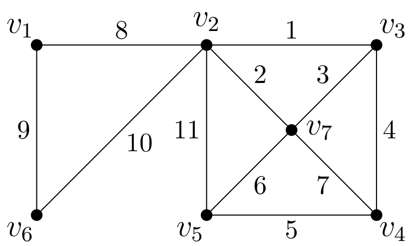
1) Marquer arbitrairement un sommet.
2) Poser $F = \varnothing$.
3) Tant qu'il existe un sommet non marqué, choisir une arête de poids minimum joignant un sommet marqué à un sommet non marqué ; ajouter cette arête à F et marquer ce nouveau sommet.

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.

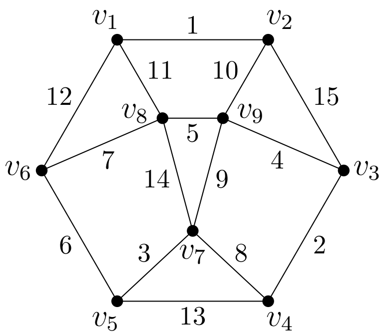
Exemple (fin) Appliquons l'algorithme de Prim aux six ordinateurs reliés par un réseau de transmission à fibre optique, en marquant initialement le sommet C_1 :



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.

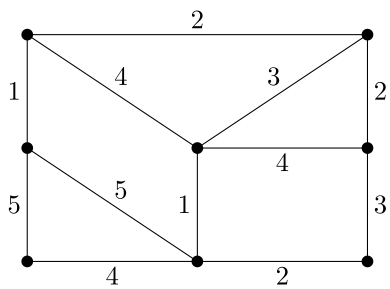
	Berlin	Londres	Madrid	Moscou	Paris	Rome
Berlin	0	7	15	11	7	10
Londres	7	0	11	18	3	12
Madrid	15	11	0	27	8	13
Moscou	11	18	27	0	18	20
Paris	7	3	8	18	0	9
Rome	10	12	13	20	9	0

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

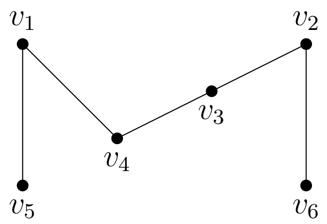
3.1 $a\,e\,f\,a\,d$ n'est pas un chemin allant de v à z.
$a\,b\,c\,d\,e$ est un chemin allant de v à u qui n'est pas simple.

3.2 1) oui 2) non

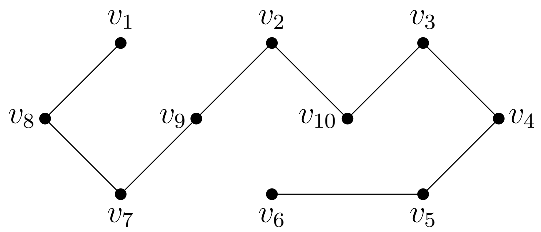
3.3 e_1 est le seul cycle de longueur 1.
e_2e_3 est le seul cycle de longueur 2.
$e_2e_4e_5$ et $e_3e_4e_5$ sont les seuls cycles de longueur 3.
Remarquons que $e_3e_2e_1$ n'est pas un cycle puisqu'un sommet est répété.

3.4 $n - 1$

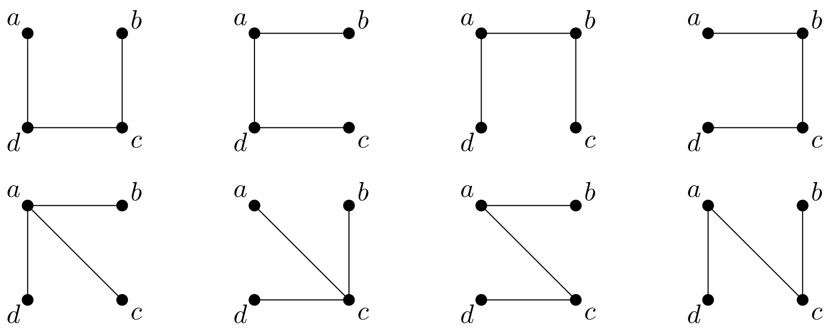
3.7



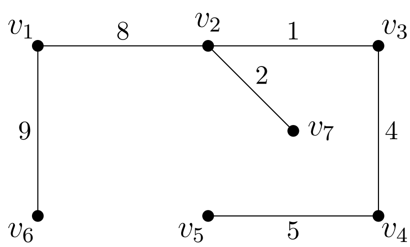
3.8



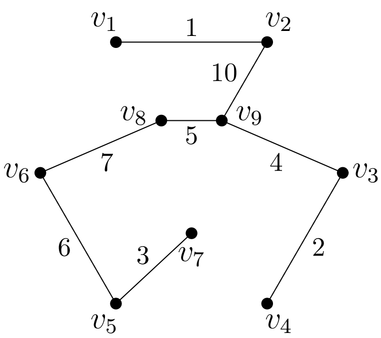
3.9



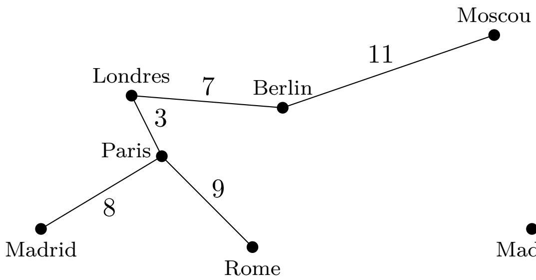
3.10

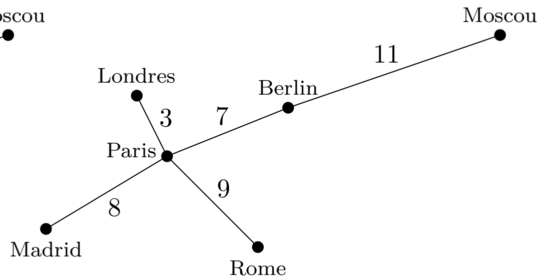


3.11

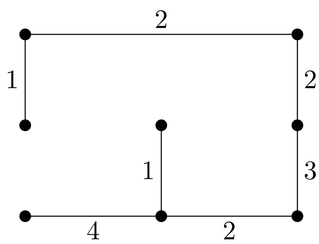

v_1
1
v_2
10
v_8
5
v_9
7
4
v_6
v_3
6
3
v_7
2
v_5
v_4

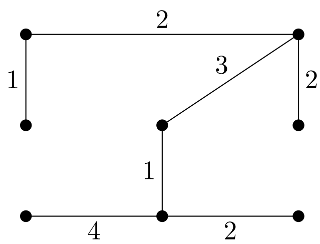
3.12


Moscou
11
Londres
7
Berlin
3
Paris
9
8
Madrid
Rome


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