CTD2 : Connexité dans les graphes

8 Connexité dans les graphes

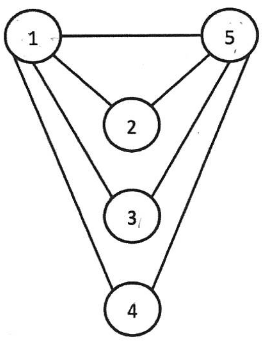
8.1 Graphes non orientés : Chaînes et cycles

Définition Une **chaîne** de longueur $q \in \mathbb{N}$, $(e_1, \ldots, e_q)$, est une séquence de q arêtes successivement adjacentes. Il existe donc une suite de sommets $(v_1, \ldots, v_{q+1})$ telle que v_i soit extrémité de e_{i-1} et e_i.

Vocabulaire
- On dit que la chaîne joint les sommets v_1 et v_{q+1}.
- Une chaîne est dite **élémentaire** si les sommets $(v_1, \ldots, v_{q+1})$ sont distincts 2 à 2.
- Une chaîne est **fermée** si $v_1 = v_{q+1}$.
- Une chaîne est **simple** si les arêtes $(e_1, \ldots, e_q)$ sont distinctes 2 à 2,
- q est la **longueur** de la chaîne.

Définitions
- Un **cycle** est une chaîne simple fermée.
- Un cycle est **élémentaire** si les sommets parcourus sont distincts 2 à 2 sauf le premier et le dernier.
- La **distance** entre deux sommets est la longueur de la plus petite chaîne entre eux.
- Le **diamètre** d'un graphe est la distance maximal entre deux sommets du graphe. Elle vaut $+\infty$ si il existe des sommets non connectés (la définition de la connexité arrive après).

Exemple/Exercice



15412531

Trouver sur ce graphe :
1 — une chaîne fermée non élémentaire ;
2 — une chaîne simple, ni fermée, ni élémentaire ;
3 — un cycle élémentaire ;
4 — un cycle non élémentaire.

1 non simple : 152152(1)
simple : 1541253(1)

2 15213

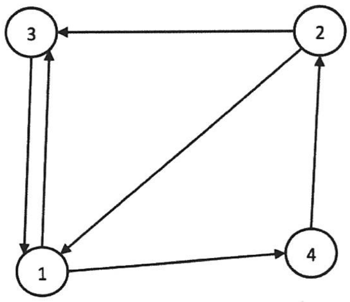
3 1521

4 15412531

8.2 Graphes orientés : Chemins et circuits

On ne parle plus de chaîne/cycle mais de chemins et de circuits qui prennent bien sûr en compte l'orientation des arcs.

Exemple/Exercice



Trouver sur ce graphe :

1 — une chemin (non fermé) élémentaire ;
2 — une chemin (non fermé) non élémentaire ;
3 — un circuit élémentaire ;
4 — un circuit non élémentaire.

1 142 ; 13 ; 1
2 1314
3 131 ; 1421
4 142131

8.3 Graphe connexe, arbre, arbre couvrant

Définitions

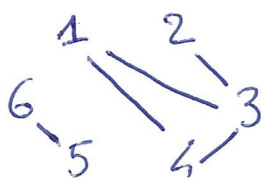
- Un graphe non orienté est **connexe**, si entre tout couple de sommet, il existe une chaîne les joignant.
- Si un graphe n'est pas connexe, il est constitué de plusieurs sous-graphes connexes appelés **composantes connexes** de ce graphe.
- Un graphe orienté est **fortement connexe** si toute paire de sommets distincts (i, j) est reliée par au moins un chemin. On appelle **composante fortement connexe** tout sous-graphe maximal fortement connexe, c'est à dire, tout sommet de cette composante a un chemin le menant à tout autre sommet, et aucun autre sommet ne peut être ajouté en préservant cette propriété.

Les composantes connexes constituent ~~des~~ une partition de V.

Exemple

1 — Dessiner un graphe à 3 composantes connexes.
2 — $G = (\{1,2,3,4,5,6\}, \{\{1,3\}, \{1,4\}, \{2,3\}, \{3,4\}, \{5,6\}\})$ est-il connexe ?
3 — Les composantes connexes sont les classes d'équivalence de quelle relation ?

1 

2 

Non pas connexe car (5,6) pas relié aux autres
Compo connexes : $\{5,6\}$; $\{1,2,3,4\}$.

3 $i \equiv j \triangleq \exists$ chemin entre i et j (relation d'équ).

Calcul des Composantes Fortement Connexes (Demoucron)

Hypothèses :

- Soit $G = (V, E)$ un graphe orienté
- On cherche à calculer CFC_G, une partition de V
- On note les successeurs et prédécesseurs :

$$\begin{aligned} Succ(S \subseteq V) &= \{v' \mid v \in S, (v, v') \in E\} \\ Pred(S \subseteq V) &= \{v' \mid v \in S, (v', v) \in E\} \end{aligned}$$

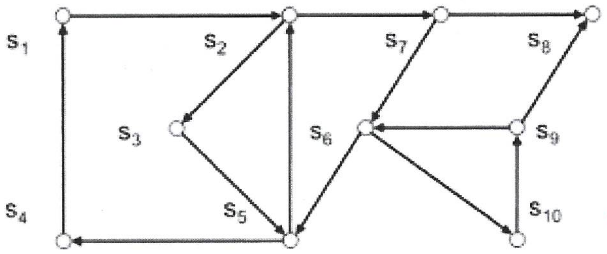
- Ainsi que leur fermeture réflexive et transitive :

$$\begin{aligned} Succ^*(S \subseteq V) &= S \cup Succ^*(Succ(S))) \\ Pred^*(S \subseteq V) &= S \cup Pred^*(Pred(S))) \end{aligned}$$

Algorithme :

1. $CFC_G \leftarrow \emptyset$
2. Choisir $v \in V$ n'apparaissant pas dans CFC_G
3. $CFC_v \leftarrow Succ^*(\{v\}) \cap Pred^*(\{v\})$
4. $CFC_G \leftarrow CFC_G \cup \{CFC_v\}$
5. Recommencer en 2 tant que possible

▷ **Exercice 11** *Appliquer l'algorithme de Demoucron au graphe suivant :*



- Soit $s_1 \in V$

$Succ^*(s_1) = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_8, s_9, s_{10}\}$.

$Pred^*(s_1) = \{s_1, s_2, s_3, s_4, s_5, s_6, s_7, s_9, s_{10}\}$

$Succ^*(s_1) \cap Pred^*(s_1) = Pred^*(s_1) = CFC_{s_1}$

- Soit $s_8 \in (V \setminus CFC_{s_1})$.

$Succ^*(s_8) = \{s_8\}$

$Pred^*(s_8) = V$

$Succ^*(s_8) \cap Pred^*(s_8) = \{s_8\} = CFC_{s_8}$

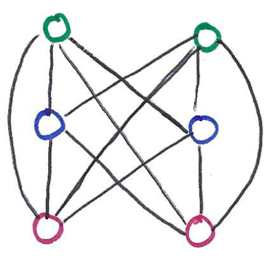
- $(V \setminus CFC_{s_1} \setminus CFC_{s_8}) = \emptyset$

$$\boxed{CFC_G = \{CFC_{s_1}, CFC_{s_8}\}}$$

▷ **Exercice 12** *Trois pays envoient chacun à une conférence deux espions ; chaque espion doit espionner tous les espions des autres pays (mais pas son propre collègue).*

- *Représentez cette situation par un graphe d'ordre 6 dans lequel chaque arête reliant i et j signifie que j espionne i et i espionne j.*

1. *Ce graphe est-il complet ? est-il connexe ?*
2. *Quel est le degré de chaque sommet ? Déduisez-en le nombre d'arêtes.*



1. Pas complet car deux espions du même pays ne s'espionne pas entre eux.
Graphe connexe car chaque sommet est relié à 4 autres et il existe 1 chemin entre ~~son~~ voisin et son camarade.

2. Chaque sommet a un degré de 4.

$$N_{A_{rêtes}} = \frac{N_{sommet} \times degré_{sommet}}{2} = \frac{6 \times 4}{2} = 12.$$

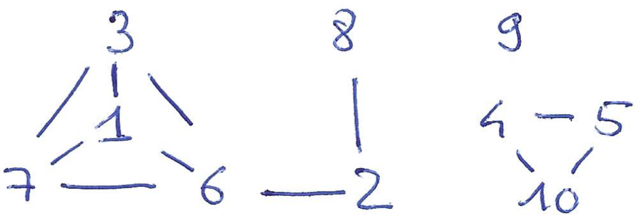
▷ **Exercice 13** *Etant donné un groupe de 10 personnes, le tableau suivant indique les paires de personnes qui sont amis.*

i	1	2	3	4	5	6	7	8	9	10
Amis de i	3,6,7	6,8	1,6,7	5,10	4,10	1,2,3,7	1,3,6	2		4,5

— *Représentez cette situation par un graphe d'ordre 10 dans lequel une arête entre les sommets i et j signifie qu'il y a une relation d'amitié entre i et j.*

① *Ce graphe est-il complet ? connexe ?*

② *Si l'adage "les amis de mes amis sont mes amis" était vérifié, que pourrait-on en conclure sur la structure du graphe ?*



① Non pas complet car pas de chemin entre 8 et 9

Connexe non car 3 gpes distincts

② Si la relation arête est transitive (ppte vraie).

=> toute composante connexe est 1 $\boxed{\text{sous graphe complet}}$ clique

Arbres couvrants, Arbres

Définitions

— Un **arbre** est un graphe connexe sans cycle ;

— une **forêt** est un graphe dont chaque composante connexe est un arbre.

▷ **Exercice 14** *Montrer l'équivalence des propositions suivantes pour un graphe G connexe*

1. *$G = (V, E)$ est un arbre ;*
2. *G est sans cycle et a $n - 1$ arêtes ;*
3. *G est sans cycle et a un cycle dès qu'on ajoute une arête ;*
4. *G est connexe et non connexe dès qu'on enlève une arête ;*
5. *tout couple de sommet est relié par une chaîne unique.*

connexe : au moins un chemin entre toute paire de sommets

sans cycle : au plus 1 chemin ___

$1 \Leftrightarrow 3$ car on rajoute 1 chemin entre deux arêtes avec déjà un chemin

$1 \Leftrightarrow 4$ si on enlève une arête, on supprime un chemin entre 2. plus connexe.

$1 \Leftrightarrow 5$

$2 \Rightarrow 1$ évident car G sans cycle et connexe donc c'est un arbre.

$1 \Rightarrow 2$ + cas initial $\#E = 0$

G connexe donc $\#V = 1$. → G sans cycle et $\#E = \#V - 1$.

+ hérédité : on suppose la ppté vraie pour tout graphe tq $\#E \leq m$.

Soit $G = (V, E)$ tq $\#E = m+1$.

Soit G_1, G_2 resp les sous-graphes des sommets reliés à v_1 / v_2 "en premier"

G_1, G_2 sont connexes sans cycle car ss graphes de G et vérifient la ppté de rec

$\#E_1 \leq m$ et $\#E_2 \leq m$

donc $\#E_1 = \#V_1 - 1$ $\qquad$ $\#E_1 + \#E_2 = \#V - 2$

$\#E_2 = \#V_2 - 1$ $\qquad$ $\#E - 1 = \#E_1 + \#E_2 = \#V - 2$

$\#E = \#V - 1$

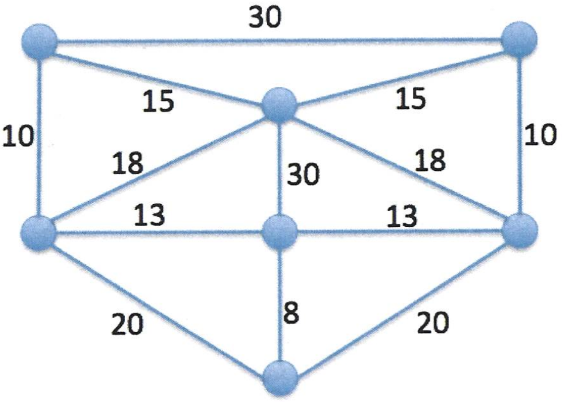
Définition Un graphe partiel connexe qui est un arbre est appelé **graphe couvrant** ou **arbre couvrant** (*spanning tree* en anglais).

→ dans le sens des poids mini

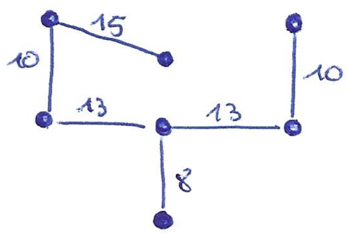
Calcul d'un arbre couvrant minimum : algorithme de Kruskal Trier les arêtes par ordre croissant de poids : on choisit les arêtes dans cet ordre, et on garde la suivante tant qu'elle ne crée pas de cycle.

Remarque C'est un algorithme glouton optimal.

Trouver un arbre couvrant minimum dans ce graphe pondéré.

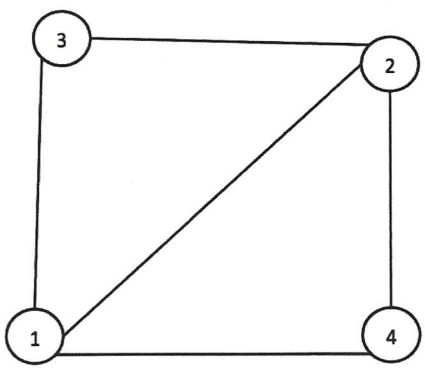


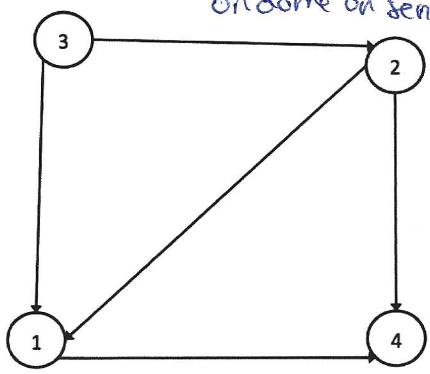
arbre couvrant minimal.



Cycles, base de cycles, nombre cyclomatique Soit G le graphe orienté suivant :

on donne un sens arbitraire.





On considère un cycle c_1 du graphe non orienté induit par G, en considérant les arcs comme des arêtes, passant par les sommets $(3, 2, 4, 1)$, c'est-à-dire, le cycle passant par les arêtes $((3, 2), (2, 4), (1, 4), (3, 1))$. On

3

$$\begin{array}{c} (1,4) \\ (2,1) \\ (2,4) \\ (3,1) \\ (3,2) \end{array} \begin{pmatrix} -1 \\ 0 \\ 1 \\ -1 \\ 1 \end{pmatrix}$$

remarque que les arcs $(1,4)$ et $(3,1)$ sont parcourus à l'envers. Si on 'range' les 5 arêtes de G dans l'ordre lexicographique, on peut exprimer le cycle c_1 comme un vecteur sur E : $c_1 = (-1, 0, 1, -1, 1)$, appelé vecteur cycle. Ce vecteur est défini à un coefficient -1 près.

De la même façon, on peut définir les autres vecteurs cycles du graphe :
$c_2 = (0, 1, 0, -1, 1)$, 3213
$c_3 = (-1, -1, 1, 0, 0)$ 4124

▷ **Démo 1** *Justifier que la somme de 2 cycles (qui partage un noeud) reste un cycle.*

On voit que $c_2 + c_3 - c_1 = (0, 0, 0, 0, 0)$. Les cycles sont donc linéairement dépendants.

Définition On appelle une **base de cycles**, un ensemble de cycles linéairement indépendants et générateur. Le **nombre cyclomatique**, noté $\nu(G)$ est le nombre d'éléments dans une base de cycles.

Théorème $\nu(G) = m - n + p$ où $m = \#E$, $n = \#V$ et p est le nombre de composantes connexes.

Exemple Trouver plusieurs bases de cycle pour le graphe précédent.
la famille de cycles elem $\{c_i\}$. est indépendante . génératrice

$$c_i = \begin{pmatrix} \\ \\ 0 \\ \vdots \\ 1 \\ \vdots \\ 0 \end{pmatrix} \begin{matrix} \}\ \text{arêtes de l'arbre} \\ \}\ \text{arêtes supp.} \end{matrix}$$
ième

Calcul d'une base de cycles A partir d'un arbre couvrant, chaque arête qui n'appartient pas à l'arbre couvrant crée un cycle ; l'ensemble des cycles créés par les arêtes manquantes crée une base de cycles.

▷ **Démo 2** *Montrer que les cycles ainsi trouvés sont indépendants, et montrer qu'il y en a $\nu(G)$.*

▷ **Démo 3** *En déduire la démonstration du théorème précédent.*

$\nu(G) = \#E + \#V + p$.
on peut raisonner sur chaque composante connexe indépendamment
On se ramène à $p = 1$ (graphe connexe).
L'arbre couvrant contient $\#V - 1$ arêtes
$\#E = \#\text{arêtes arbre couvrant} + \underbrace{\nu(G)}_{\#\{c_i\}}$.
$\#E = \#V - 1 + \nu(G)$
$\nu(G) = \#E - \#V + 1$ avec $p = 1$.

▷ **Exercice 15** *Application de l'algorithme de Kruskal (suite).*
Trouver une base de cycles dans le graphe de l'exercice précédent.