TD 1 – Arbres de poids minimum et Plus courts chemins

Questions de cours : Arbre/Forêt
Considérons un graphe connexe sans cycle (donc 1-graphe).
On appelle forêt un graphe dont chaque composante connexe est un arbre.
Soit G un graphe comptant au moins $n \geq 2$ sommets. Montrer que les propriétés suivantes sont équivalentes pour que G soit un arbre :

a) G est connexe est sans cycle.
b) G est sans cycle et compte n-1 arêtes.
c) G est connexe et compte n-1 arêtes.
d) G est sans cycle est en ajoutant une arête quelconque on crée un cycle.
e) G est connexe et si on supprime une arête quelconque, il n'est plus connexe.
f) Tout couple de sommets de G distincts est relié par une chaîne unique.

Corrigé : voir corrigé manuscrit

a ⇒ b : Notons p le nombre de composantes connexes de G (p=1) et m le nombre d'arêtes. Puisque pas de cycles : v(G)=0 ; or le nombre cyclomatique v(G)=m – n + p. D'où m=n-1

b ⇒ c : v(G)=0, m=n – 1 ⇒ p=1

c ⇒ d : p=1, m=n – 1 ⇒ v(G) =0
Dans la formule de v, si on ajoute une arête, seul m croît car p qui valait 1 reste à 1.
Soit G' le graphe obtenu.
On a : v(G')=1 : il existe un cycle

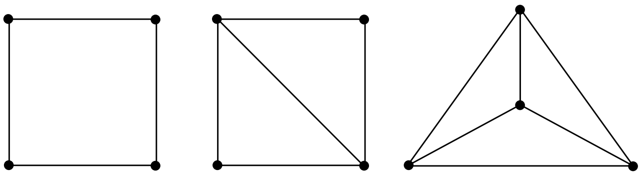
d ⇒ e : Si G n'était pas connexe : il existerait deux sommets a et b non connectés ; si on ajoute une arête {a,b} on ne peut créer de cycle, contradiction, donc G connexe, soit p=1.
Vu que V(G)=0, m = n – 1. Si on supprime une arête : m' = n – 2. Le graphe G' obtenu est toujours dans cycles, donc v(G')=0, donc p'=2.

e ⇒ f : Vu que G connexe, ∀ a, b sommets il existe une chaine de a vers b. S'il en existait deux, la suppression d'une arête n'appartenant qu'à l'une d'elles ne déconnecterait pas G.

f ⇒ a : Si G admet un cycle, au moins un couple de sommets serait relié par deux chaines distinctes.

Exercice 1 (Arbres)

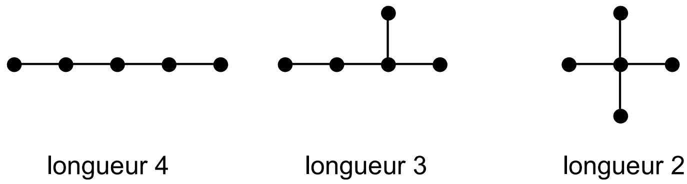
– Combien d'arbres différents existe-t-il avec 5 sommets ? avec 6 sommets ? avec 7 sommets ?



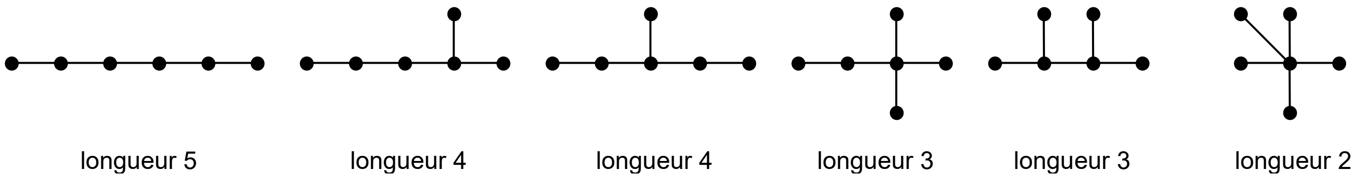
Corrigé :

Il est possible de classer ces différents arbres selon le nombre de feuilles (sommets sans fils) par exemple. Une façon de procéder peut-être plus « visuelle » consiste à les classer selon la longueur du plus long chemin dans l'arbre (on cherche ensuite à rajouter les « sommets manquants », sans les raccrocher aux extrémités du chemin pour ne pas le rallonger…). Nous obtenons ainsi :

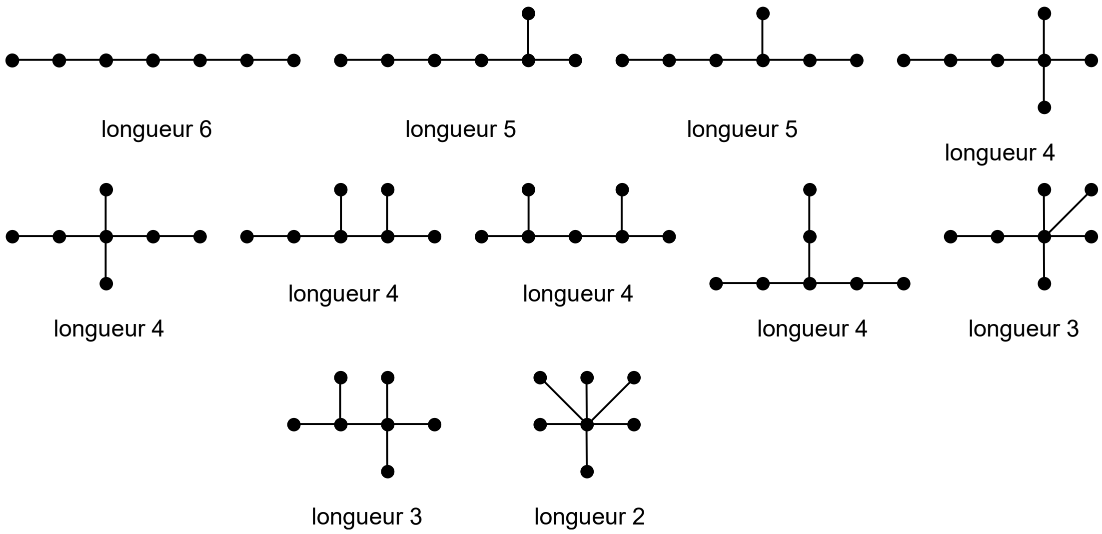
- 3 arbres à 5 sommets :



- 6 arbres à 6 sommets :



- 11 arbres à 7 sommets :

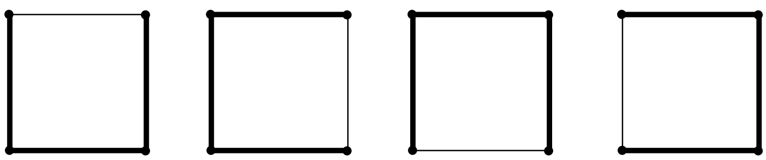


– Combien d'arbre couvrants différents les graphes suivants possèdent-ils ?

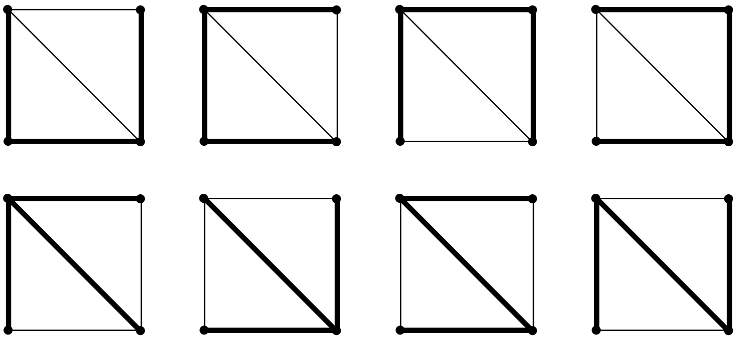
Corrigé :

Il suffit de prendre le temps de les énumérer…

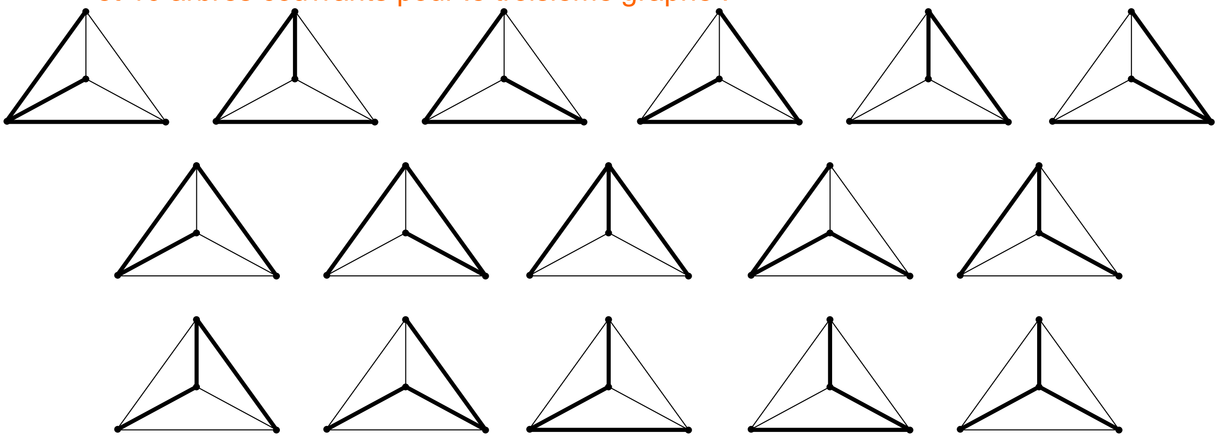
- nous obtenons 4 arbres couvrants pour le premier graphe :



- 8 arbres couvrants pour le deuxième graphe :

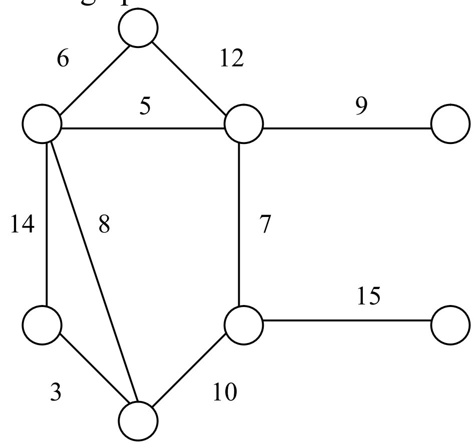


- et 16 arbres couvrants pour le troisième graphe :



Exercice 2 (Algorithme de PRIM)
Soit $G = (V, E)$ un graphe connexe et $w : E \rightarrow \mathrm{IR}^+$ une fonction de poids. On suppose que G est codé par listes de voisins : à chaque sommet v de G est associée la pile $Vois(v)$ des voisins de v. L'ensemble des sommets de V est $\{x_1, \ldots, x_n\}$.

Soit le graphe suivant :



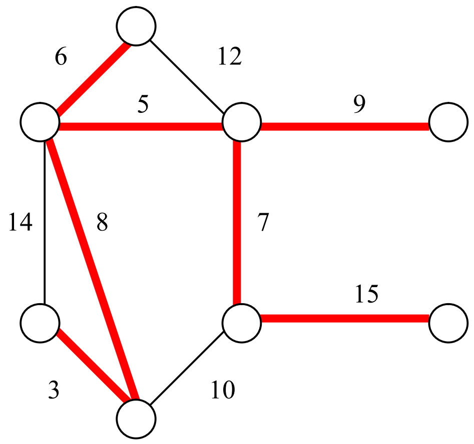
a. Selon vous, quelles arêtes appartiennent sûrement à l'arbre recouvrant minimal ?
b. Retrouvez l'arbre maximal de coût minimal par les deux algorithmes de PRIM et de KRUSKAL.

Corrigé :

Voici l'ordre des arêtes ajoutées à l'arbre par les deux algorithmes.

PRIM : 15, 7, 5, 6, 8, 3, 9 (on part d'un sommet quelconque, ici en bas à droite)
KRUSKAL : 3, 5, 6, 7, 8, 9, 15

Le résultat est :

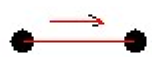
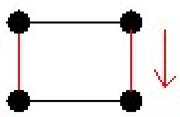
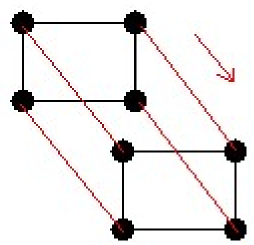


b. Soit d un entier positif non nul. L'*hypercube* Q_d est le graphe dont l'ensemble des sommets est l'ensemble des d-uplets $(x_1, . . . , x_d)$ de 0 et de 1, deux d-uplets étant adjacents s'ils diffèrent sur une seule entrée.

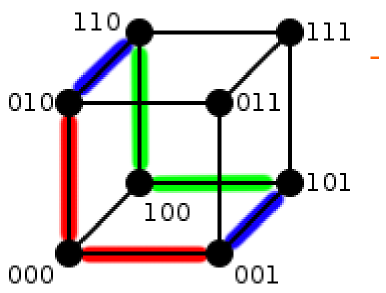
- Dessiner Q_d, pour $d = 1, . . . , 3$.
- Pour d quelconque, déterminer le nombre de sommets et d'arêtes de Q_d.
- On considère la fonction de poids p définie sur les arêtes de Q_d qui vérifie $p(e) = i$ lorsque i est l'indice de la coordonnée qui diffère entre les extrémités de e. Déterminer un arbre couvrant de poids minimum de $Q3$.
- Tester le deuxième algorithme sur le graphe $Q3 \setminus \{111\}$, pondéré comme dans la question précédente.

Corrigé : voir corrigé manuscrit

Q1 Q2 Q3

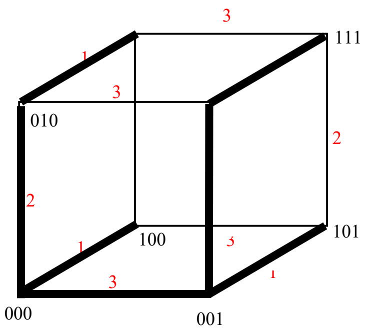




- Le nombre de sommets est : 2^d
- Le nombre d'arêtes est : $2^{(d-1)}*d$



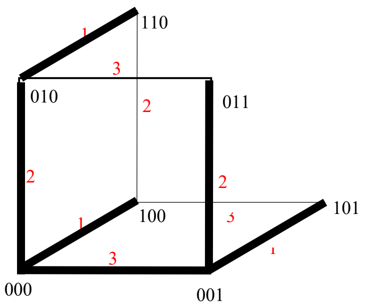
Pour déterminer l'arbre couvrant de poids minimal on utilise l'algorithme de Kruskal, le poids des arêtes étant l'indice par lequel diffèrent les deux sommets qui sont reliés par une arête. Par exemple le poids de l'arête 010 et 000 est 2 car l'indice qui diffère est le deuxième, celui de l'arête 101 et 001 est 1 et ainsi de suite.

En appliquant l'algorithme de Kruskal sur le cube de la figure précédente, on obtient l'arbre couvrant de poids minimal représenté en tirets fins.



Cout (A) = 11

En appliquant une deuxième fois l'algorithme de Kruskal sur Q3\{111} nous avons : l'arbre couvrant de poids minimal représenté en tirets fins



Cout (A') = 10

Exercice 4 (Réseau à expansion minimale - Arbre recouvrant de coût minimal)

Une banque Toulousaine, Midi Pyrénées, est en train de mettre en place un réseau pour connecter ses sept agences au siège central à Toulouse. Les sept villes dans lesquelles les agences sont localisées et la matrice des coûts sont indiquées dans le tableau 1 (les coûts sont calculés en fonction des offres des opérateurs et sont proportionnels aux distances entre les agences).

1. Trouver une solution en construisant l'arbre maximal à coût minimal.
2. Le trafic généré par chaque branche (proportionnel à la population de la ville) est indiqué, en terme de messages/min, dans la table 2. Pas plus de 20 messages/min, ne peuvent être envoyés sur chaque lien. Proposer une méthode qui permet de trouver l'arbre recouvrant de coût minimal en tenant compte de ces contraintes.

	1 TOULOUSE	2 BLAGNAC	3 ALBI	4 AUCH	5 CAHORS	6 FOIX	7 RODEZ	8 TARBES
1 TOULOUSE	-	2	52	13	45	15	58	59
2 BLAGNAC	2	-	52	14	43	16	58	62
3 ALBI	52	52	-	60	85	42	23	55
4 AUCH	13	14	60	-	50	18	72	50
5 CAHORS	45	43	85	50	-	59	81	95
6 FOIX	15	16	42	18	59	-	55	41
7 RODEZ	58	58	23	72	81	55	-	78
8 TARBES	59	62	55	50	95	41	78	-

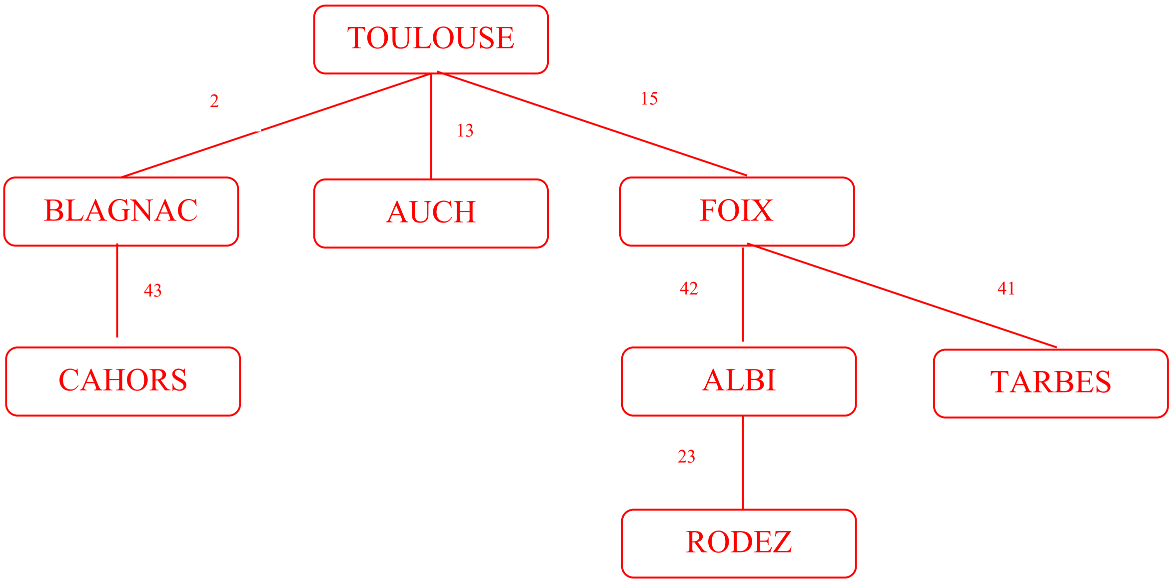
Figure 1: Matrice des coûts

	Population	Trafic (messages/min)
1 TOULOUSE	443 103	30
2 BLAGNAC	20 586	14
3 ALBI	47 800	12
4 AUCH	21 838	13
5 CAHORS	23 128	6
6 FOIX	9 109	7
7 RODEZ	23 707	6
8 TARBES	46 275	6

Figure 2: Messages générés

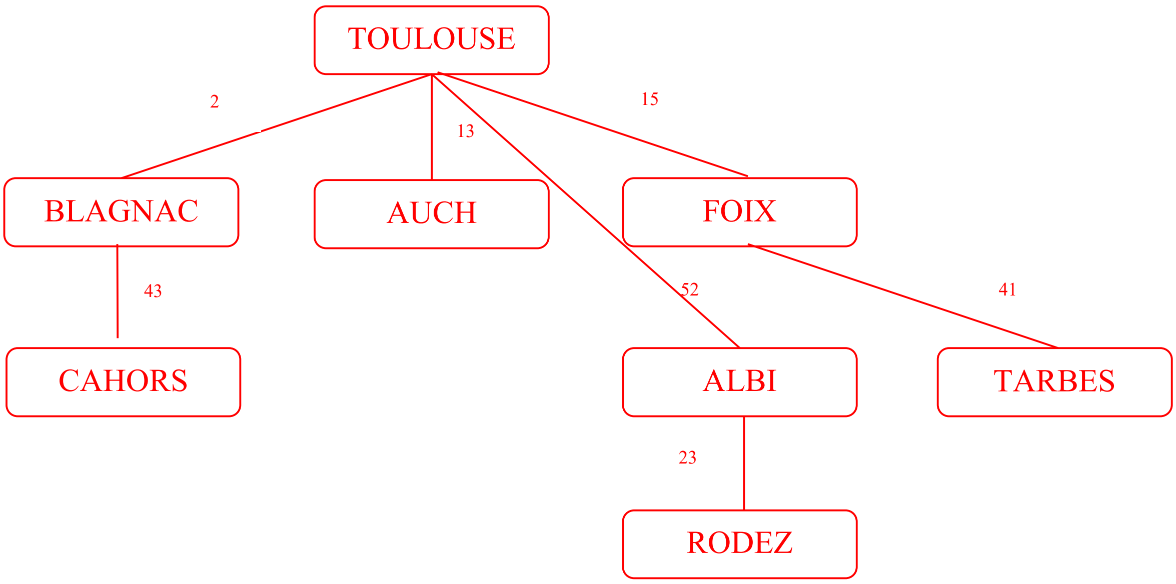
Corrigé : voir corrigé manuscrit

1) L'arbre maximal à coût minimale est :



Cout (A)=179

1) L'arbre maximal à coût minimale est :



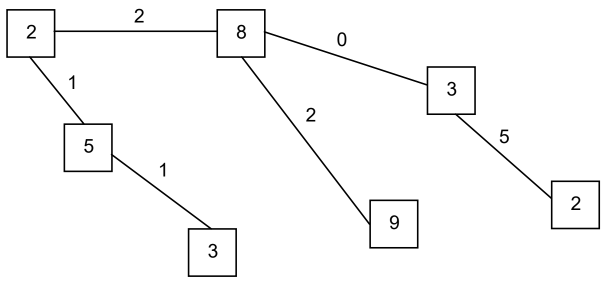
Cout (A)=189

Exercice 5 optionnel (Arbre recouvrant)

Le schéma ci-dessous représente la carte d'un groupe de villages. Les sommets sont des villages et les arêtes des chemins reliant les villages. Pour chaque village, la valeur du sommet correspond au nombre d'enfants du village en âge d'être scolarisé. Pour chaque chemin reliant deux villages, la valeur correspond au nombre de rivières que doivent traverser à la nage les piétons empruntant ce chemin.

On souhaite choisir l'un de ces villages pour y construire une école. Le critère de choix principal est la sécurité : on veut minimiser le nombre de traversées à la nage !…

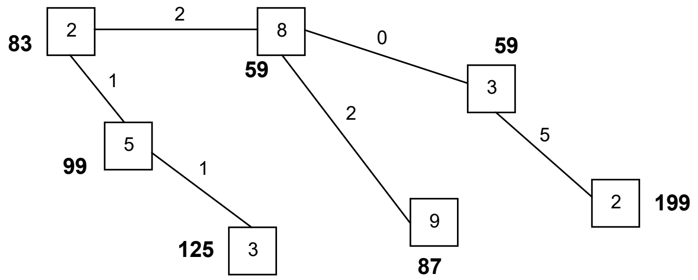
- Dans quel village doit-on construire cette école ?



Corrigé :

Il est possible, pour chacun des sommets du graphe, de calculer le nombre de traversées à la nage nécessaires si l'école est implantée dans le village correspondant… On obtient alors :

(2x8 + 2x3 + 7x2 + 4x9 + 1x5 + 2x3 = 83)



Nous avons ainsi le choix entre deux villages qui obtiennent le meilleur score (59)…