Connexité dans des graphes.

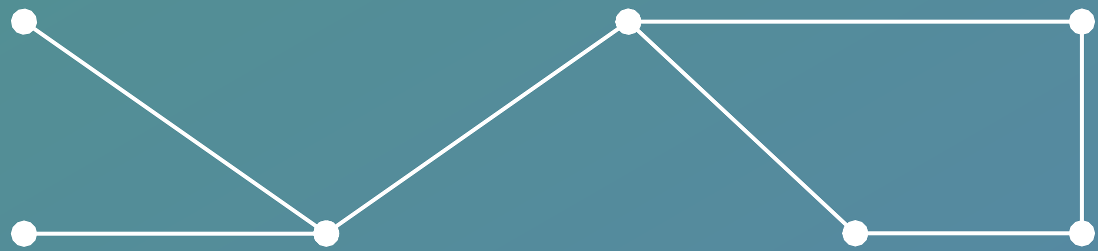
Jérémy Rouot

e-mail: jeremy.rouot@yncrea.fr

bureau 332

Un graphe $G=(X,E)$ est dit ***connexe*** s'il possède la propriété suivante :

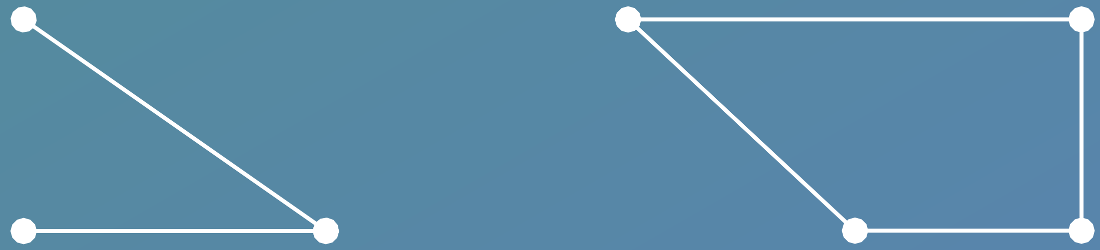
$$\boxed{\forall\ x,y\in X,\ \ x=y \text{ ou } \exists \text{ une chaîne entre } x \text{ et } y.}$$

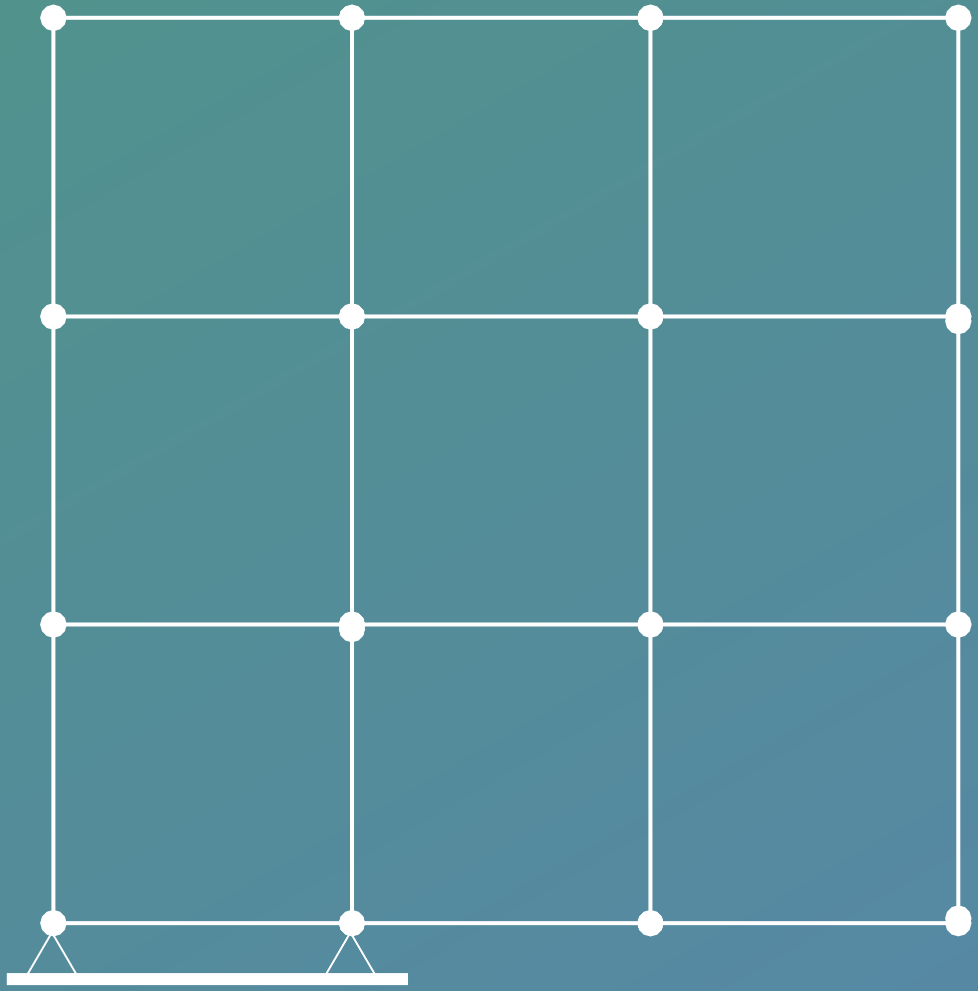


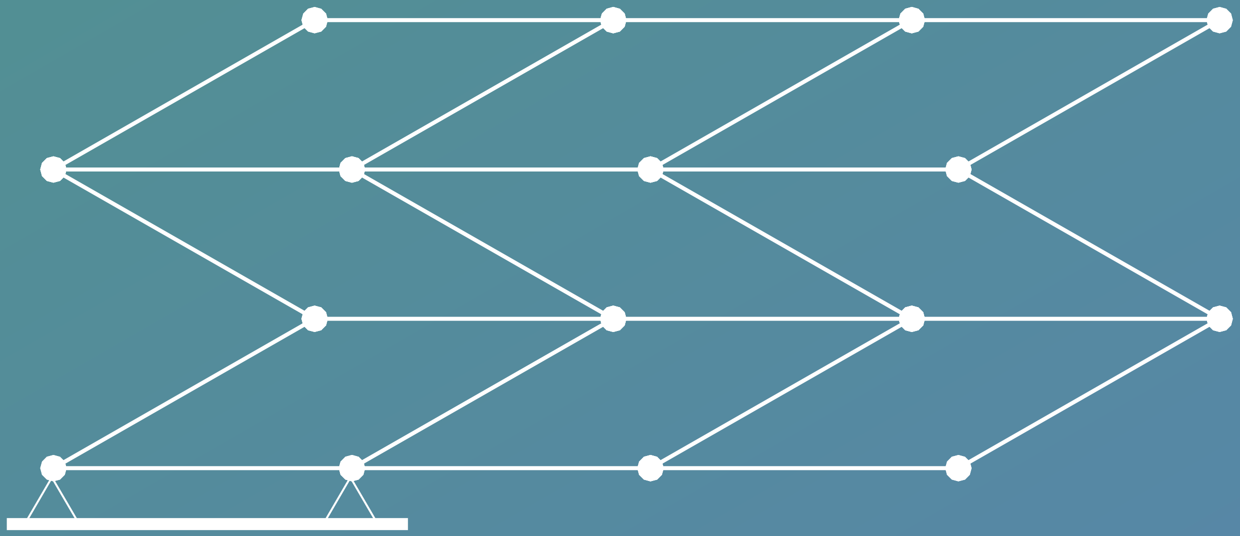
Pour un graphe quelconque la relation binaire $\mathcal{R} \subset \mathcal{X} \times \mathcal{X}$ définie par :

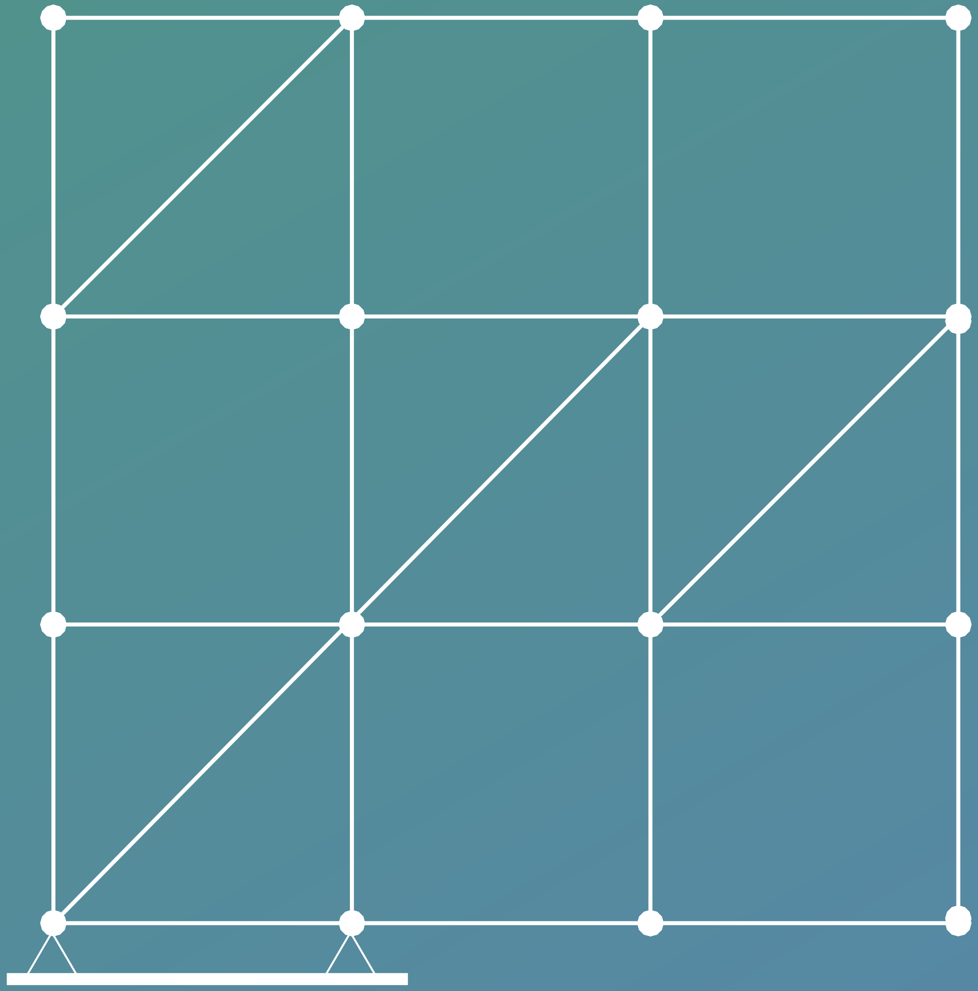
$$\boxed{x\,\mathcal{R}\,y \quad \Leftrightarrow \quad x=y \text{ ou } \exists \text{ une chaîne entre } x \text{ et } y}$$

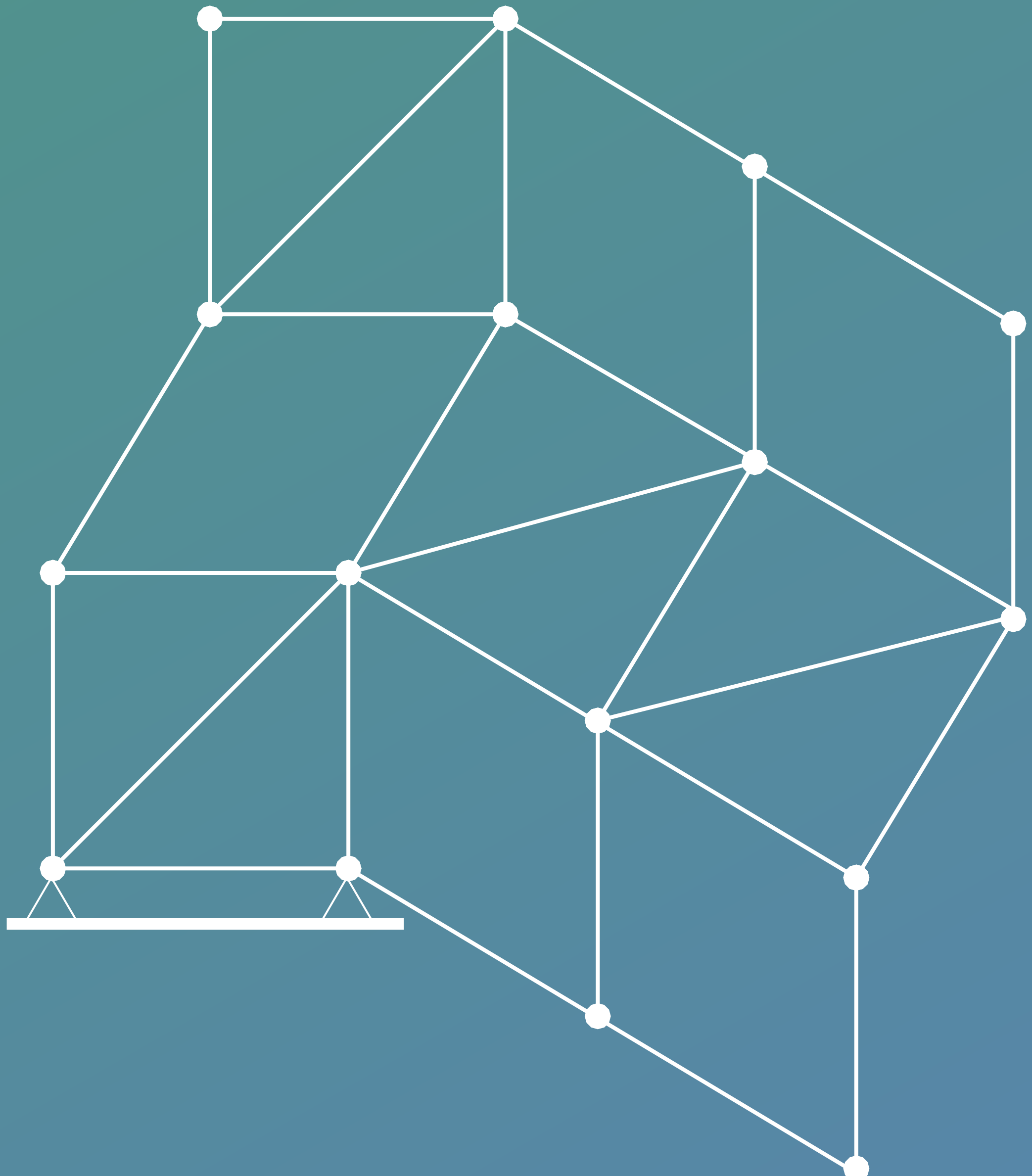
est une relation d'équivalence. Le sous-graphe de G, engendré par une classe d'équivalence de cette relation s'appelle ***composante connexe*** du graphe *G*. En d'autres mots, un graphe est connexe s'il ne possède qu'une seule composante connexe.

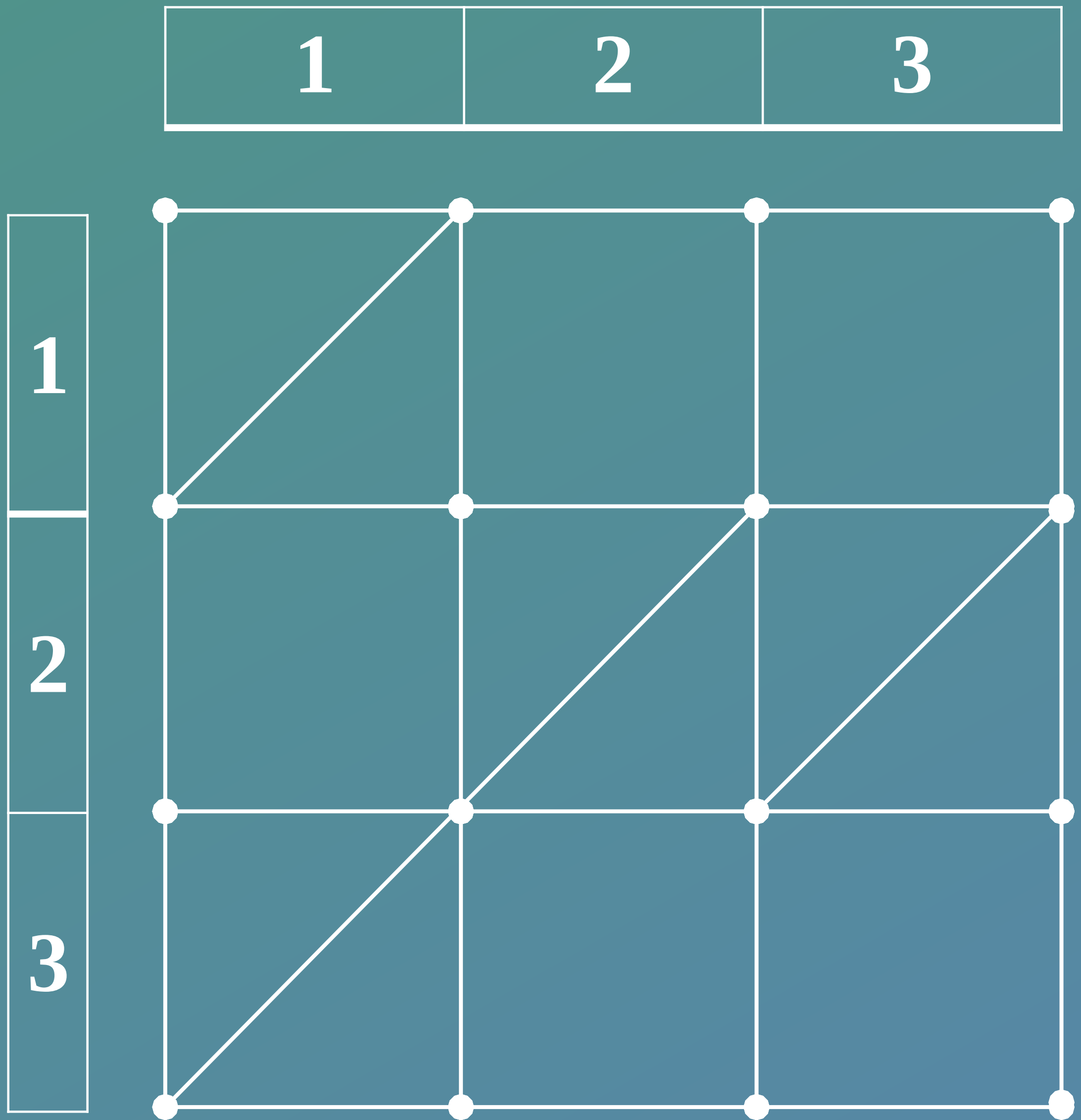


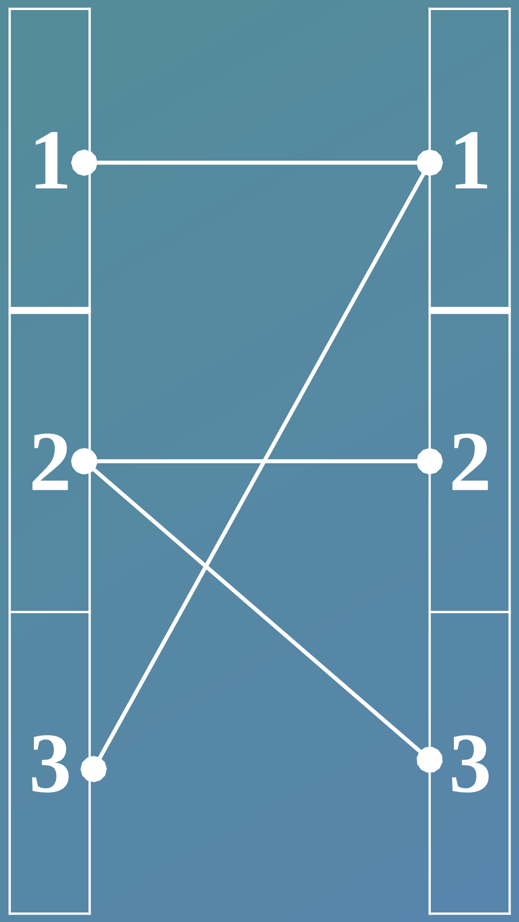




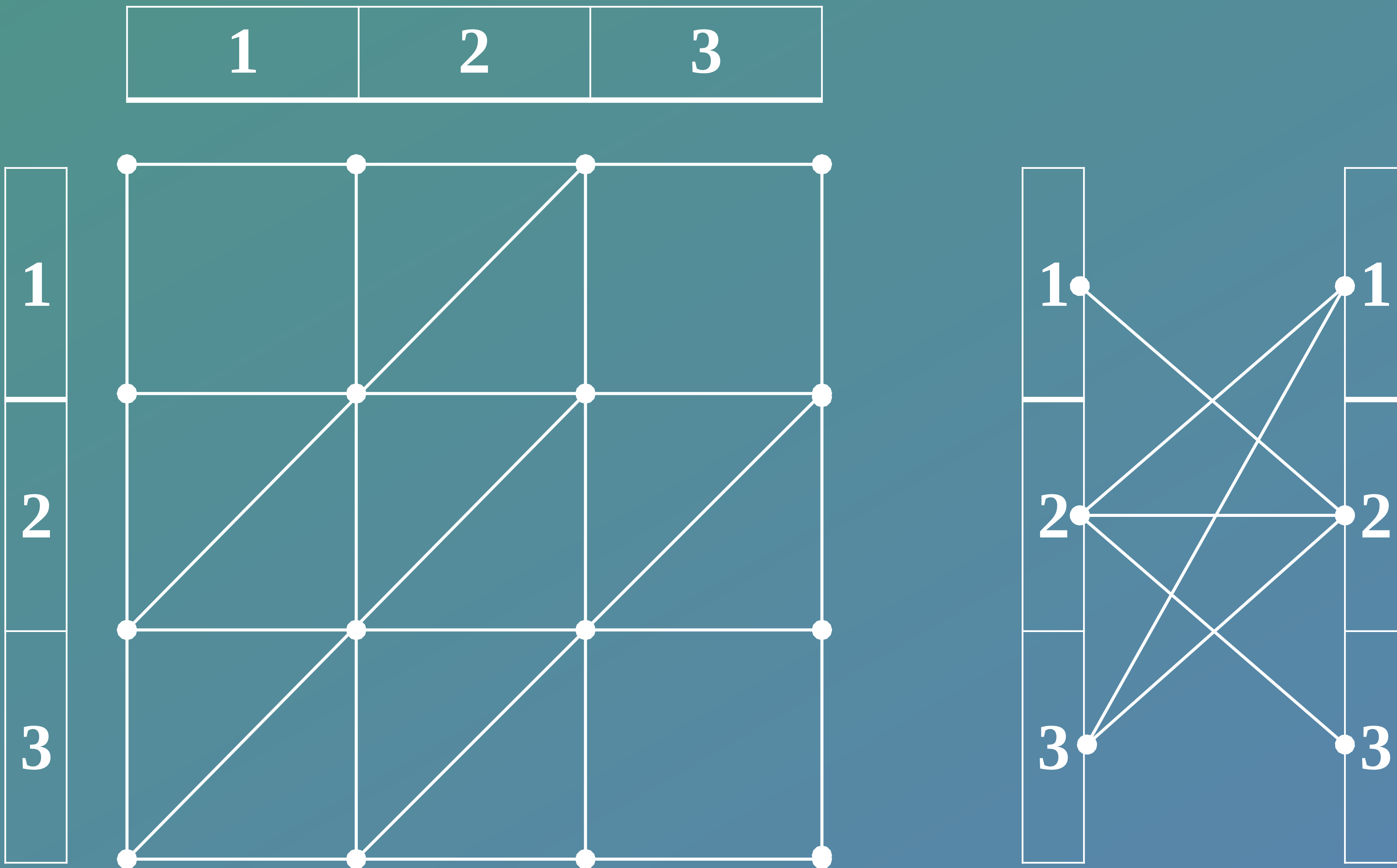


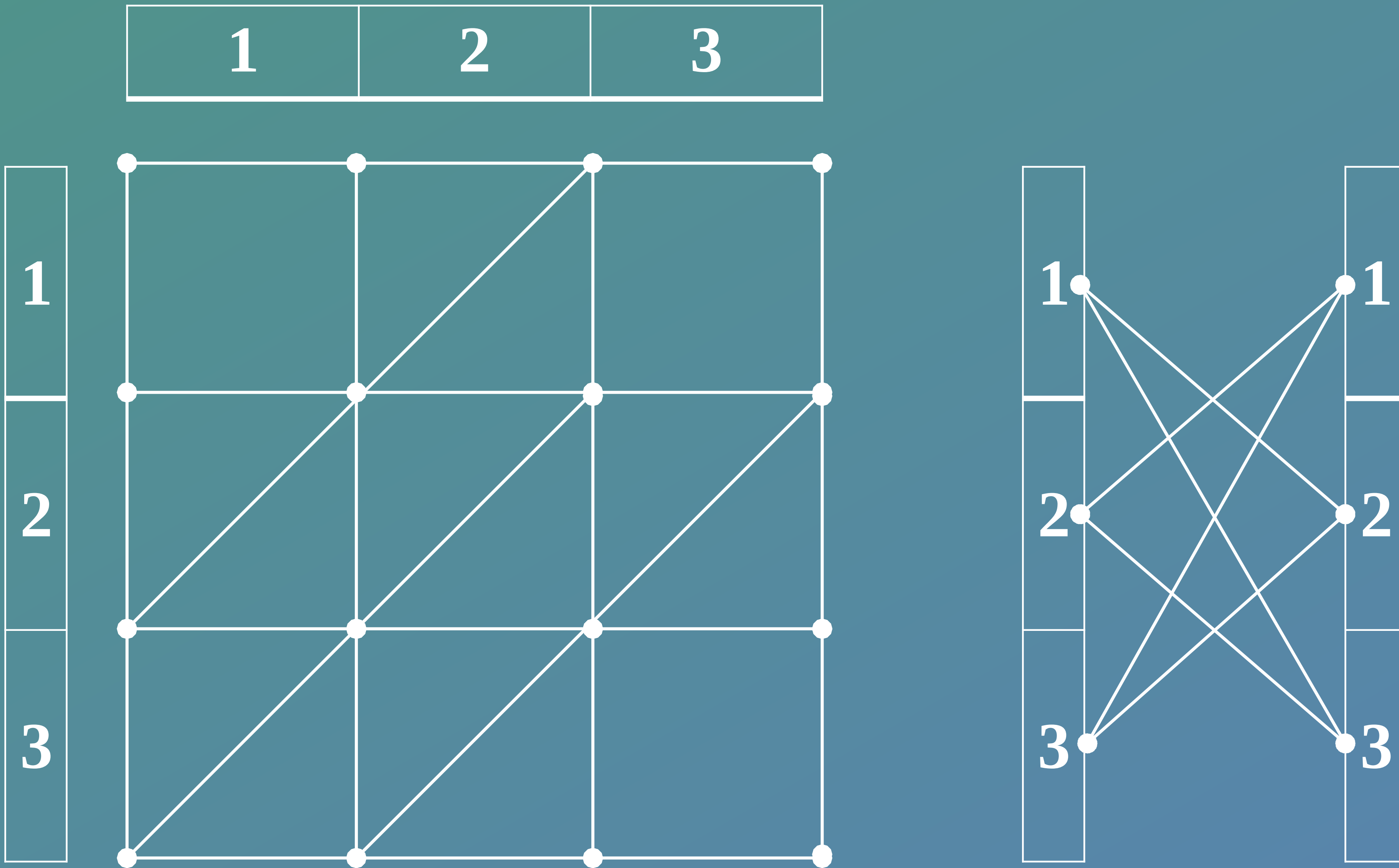




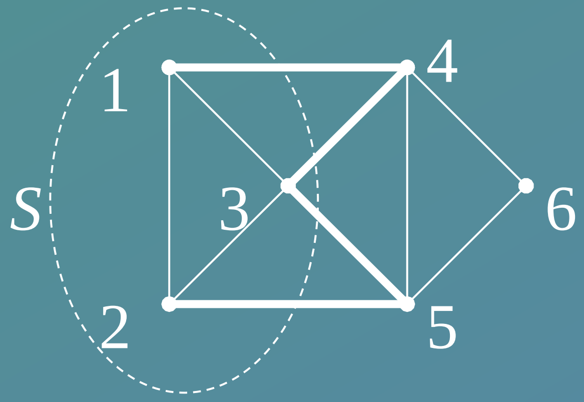


THÉORÈME (Bolker, Crapo 1977) : La grille avec $m\times n$ carrés est rigide si et seulement si le graphe associé est connexe.


1
2
3
1
2
3
1
2
3
1
2
3


1
2
3
1
2
3
1
2
3
1
2
3

La notion du ***cocycle*** permet d'étudier la connexité:



Soit S un sous-ensemble de sommets du graphe $G(X;E)$.

On appelle *cocycle associé à S*, l'ensemble $\omega_G(S)$ des arêtes de E ayant exactement une extrémité dans S.

Un sous-ensemble U d'arêtes est appelé ***cocycle*** du graphe, s'il existe une partie S de sommets telle que $U=\omega_G(S)$. Sur la figure le cocycle associé à {1, 2, 3} est {14, 34, 35, 25}. C'est aussi le cocycle associé à {4, 5, 6}.

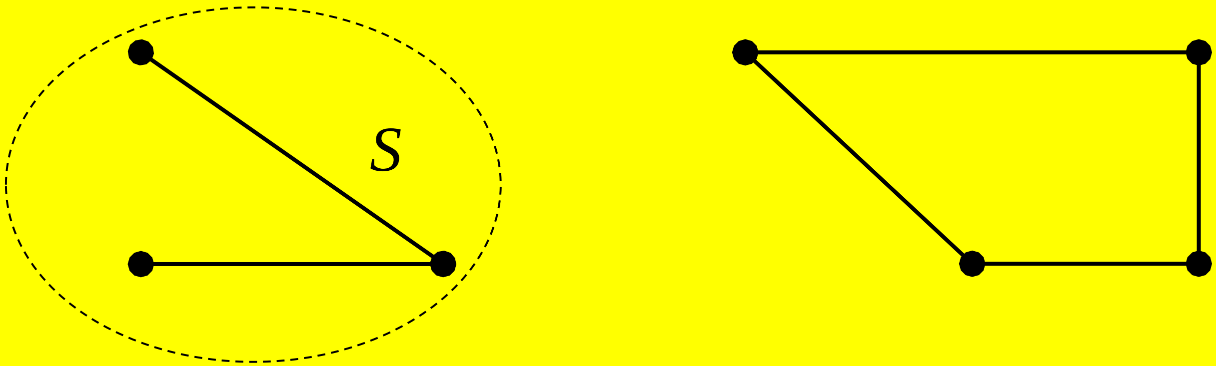
Théorème :

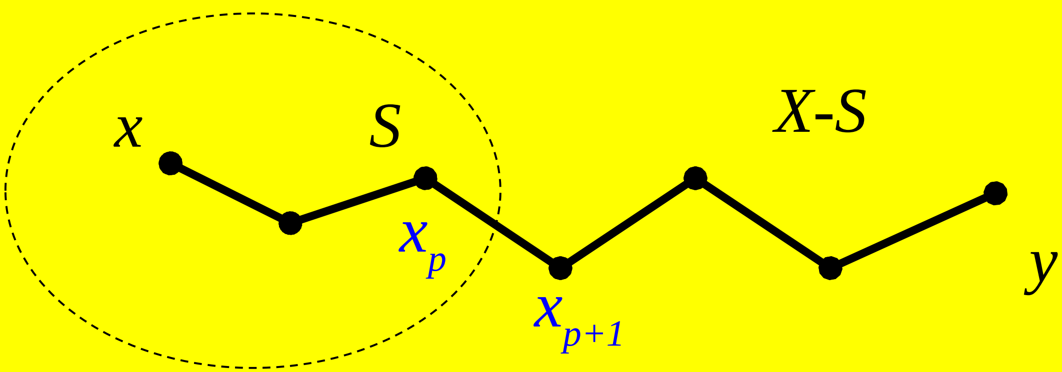
Un graphe $G=(X,E)$ est connexe

si et seulement si,

pour tout sous-ensemble de sommets S tel que $\emptyset \neq S \neq X$ on a :

$$\omega_G(S) \neq \emptyset.$$





Théorème :

Un graphe G=(X,E) est connexe
si et seulement si,
pour tout sous-ensemble de sommets S tel que
$\emptyset \neq S \neq X$ *on a :* $\boldsymbol{\omega_G(S) \neq \emptyset}$.

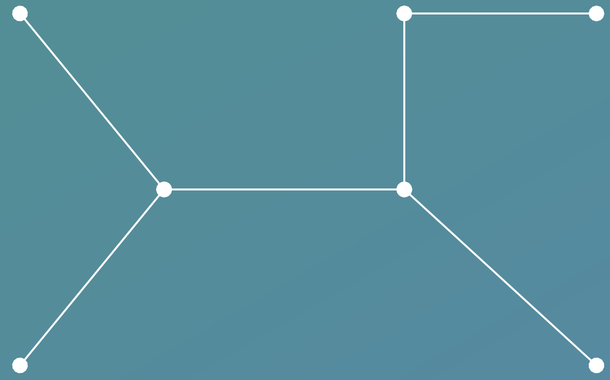
Preuve :

$\Rightarrow$ *Soit S un sous-ensemble de sommets S tel que* $\emptyset \neq S \neq X$*; soient* $x \in S$ *et* $y \in X \setminus S$*. puisque G est connexe il existe une chaîne* $\boldsymbol{\Gamma}=(x,e_1,x_1,e_2,\ldots,x_{k-1},e_k,y)$*. Soit p le plus grand indice tel que* $x_p \in S$*. On a:* $p \leq k-1$ *et* $x_{p+1} \in X \setminus S$ *donc* $e_p=\{x_p, x_{p+1}\} \in \omega_G(S)$*.*

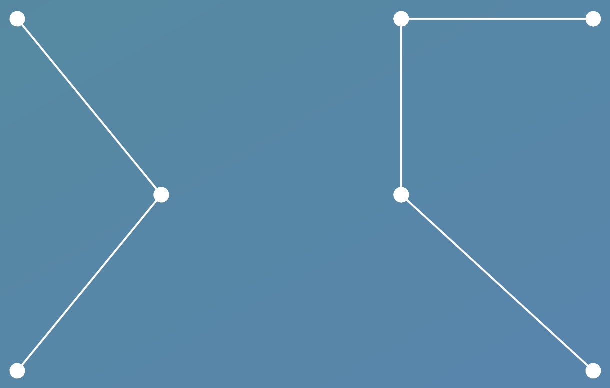
$\Leftarrow$ *Supposons que G=(X;E) n'est pas connexe. Considérons* $G_1=(X_1;E)$ *une composante connexe de G. L'ensemble* $X \setminus X_1$ *n'est pas vide et il n'y a aucune arête entre un sommet de* X_1 *et un sommet de* $X \setminus X_1$ *ce qui veut dire que* $\omega_G(X_1) = \emptyset$*.*

On appelle ***arbre*** un graphe *connexe* et *sans cycle*.

un arbre



Un graphe *sans cycle* est une ***forêt***.



une forêt de 2 arbres

Théorème :

Soit G=(X,E) un graphe d'ordre $|X| = n \geq 2$. *Les propriétés suivantes sont équivalentes:*

(1) *G est* ***connexe*** *et* ***sans cycles****;*

(2) *G est* ***sans cycles*** *et admet* ***n–1*** *arêtes;*

(3) *G est* ***connexe*** *et admet* ***n–1*** *arêtes;*

(4) *G est* ***connexe-minimal*** *(si on supprime une arête quelconque, il n'est plus connexe);*

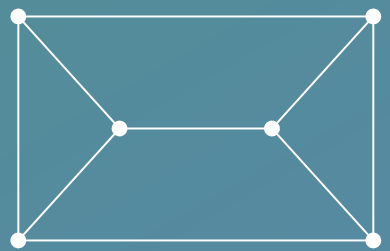
(5) *G est sans boucle et tout couple de sommets est relié par une chaîne unique.*

Un graphe $G=(X,E)$ est connexe si et seulement si il contient un graphe partiel $A=(X;T)$ qui est un arbre.

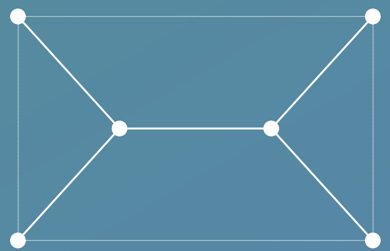
Un tel arbre est dit ***couvrant*** (car il couvre tous les sommets du graphe) ou ***arbre du graphe*** G.

Son complémentaire par rapport à G, c'est-à-dire le graphe $K=(X,E\backslash T)$, est appelé ***co-arbre associé*** à A.

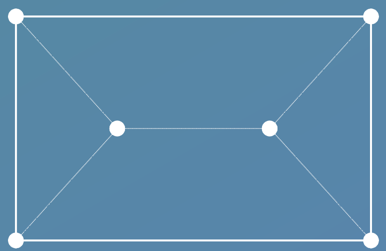
Plus généralement, on appelle co-arbre de G tout graphe partiel de G dont le complémentaire est un arbre couvrant de G.



graphe G



un arbre couvrant



co-arbre associé

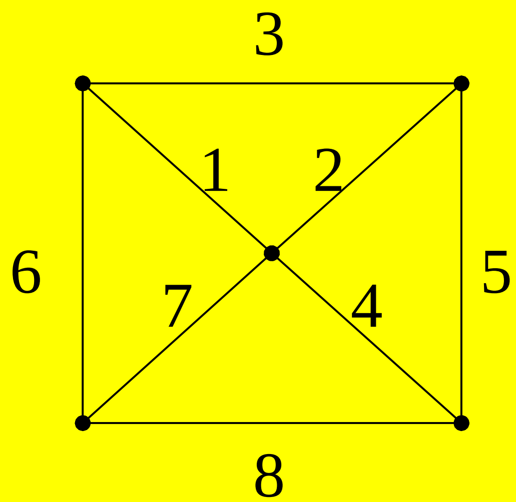
L'arbre de poids minimum ***d'un graphe connexe.***

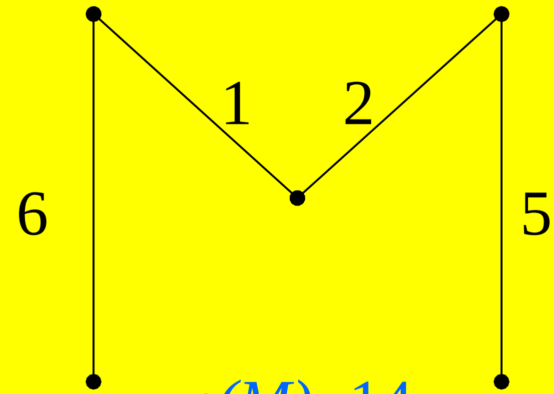
Soit $G=(X;E)$ un graphe ***connexe*** d'ordre n et $p\colon E\to R$ une fonction qui associe à toute arête de G un poids réel. Pour chaque arbre $A=(X;T)$ du G on définit son poids

$$p(A) = \sum p(e).$$

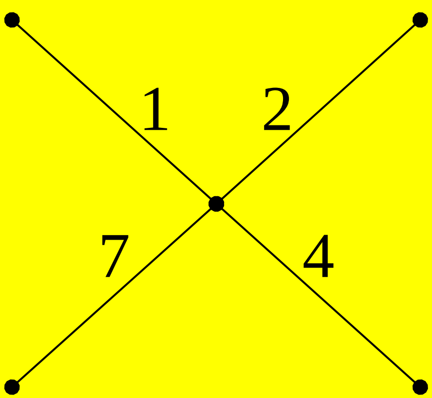
On veut déterminer un arbre couvrant de G de poids minimum.

(Le problème de maximisation peut être résolu par le remplacement de la fonction p par $-p$.)

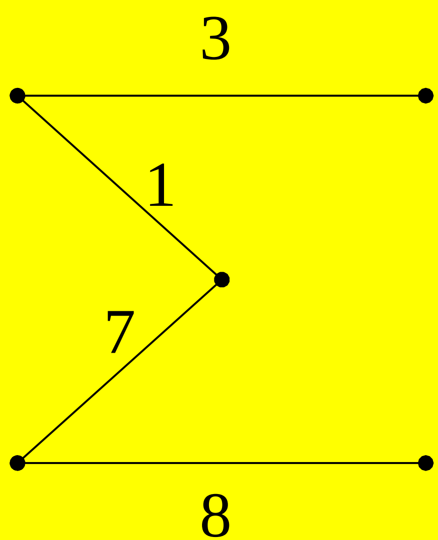




$p(M)$=14



$p(X)$=14



$p(\Sigma)$=19

Arbre de poids minimum.

Algorithme de Kruskal:

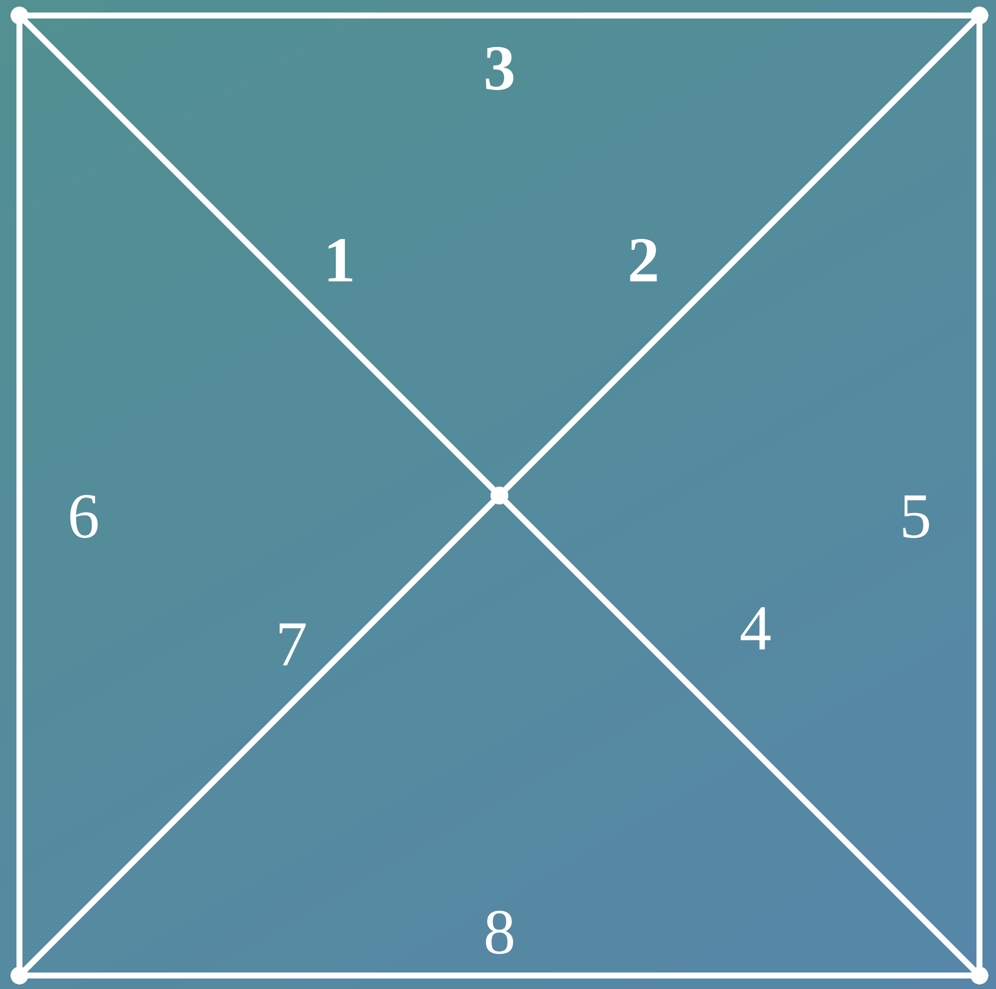
1° Trier E pour obtenir la liste $(e_1, e_2,\ldots, e_m)$ telle que :

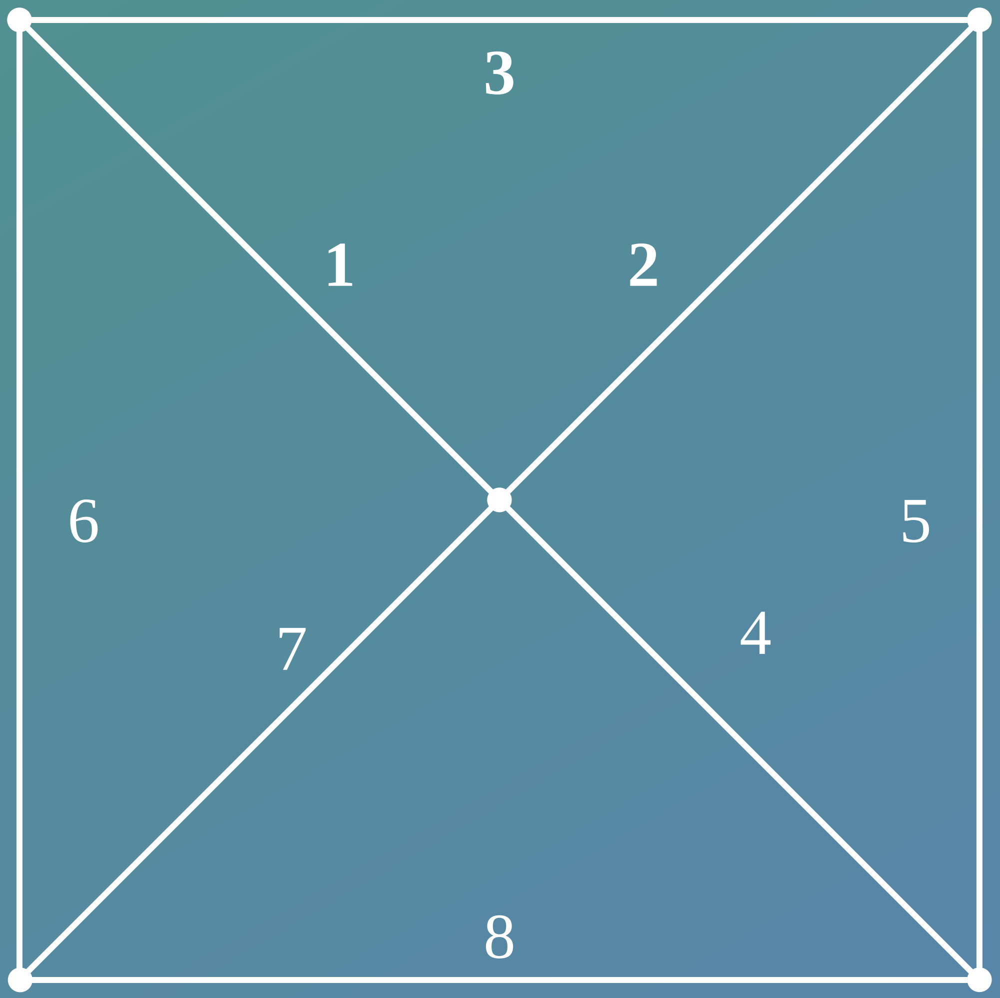
$$p(e_i) \leq p(e_{i+1}) \quad \text{pour } i=1, 2,\ldots, m-1.$$

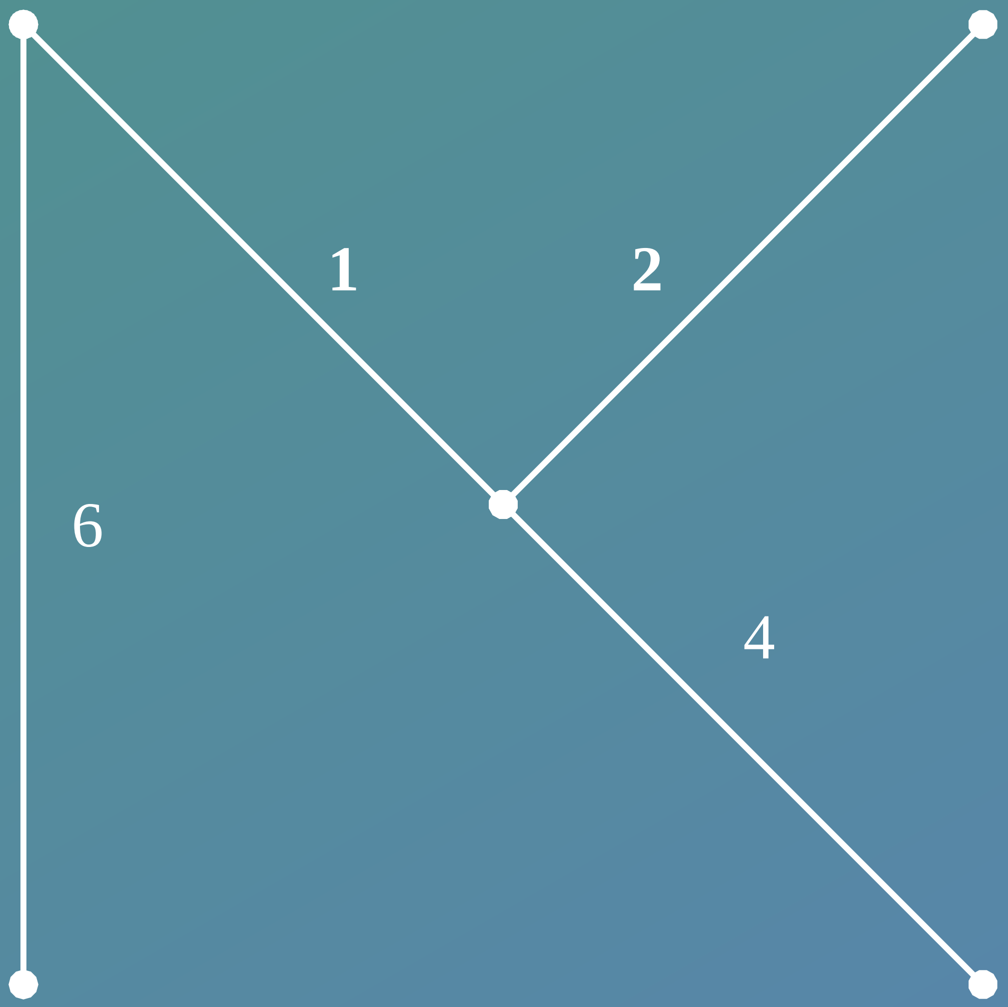
2° Construire la séquence: $\boldsymbol{T_1=\emptyset}$; $\boldsymbol{T_{i+1}=T_i \cup \{e_k\}}$

où $k = \min\{j;\ \boldsymbol{T_i+e_j} \text{ est sans cycle}\}$.

$\boldsymbol{A_n=(X;T_n)}$ est un arbre de poids minimum.


3
1
2
6
5
7
4
8


3
1
2
6
5
7
4
8


1
2
6
4

$$p(e_1) \leq p(e_2) \leq \ldots \leq p(e_{m-1}) \leq p(e_m)$$

$$O(m) \quad ?$$

NON !

$$O(m \lg m)$$

Algorithme de Prim:

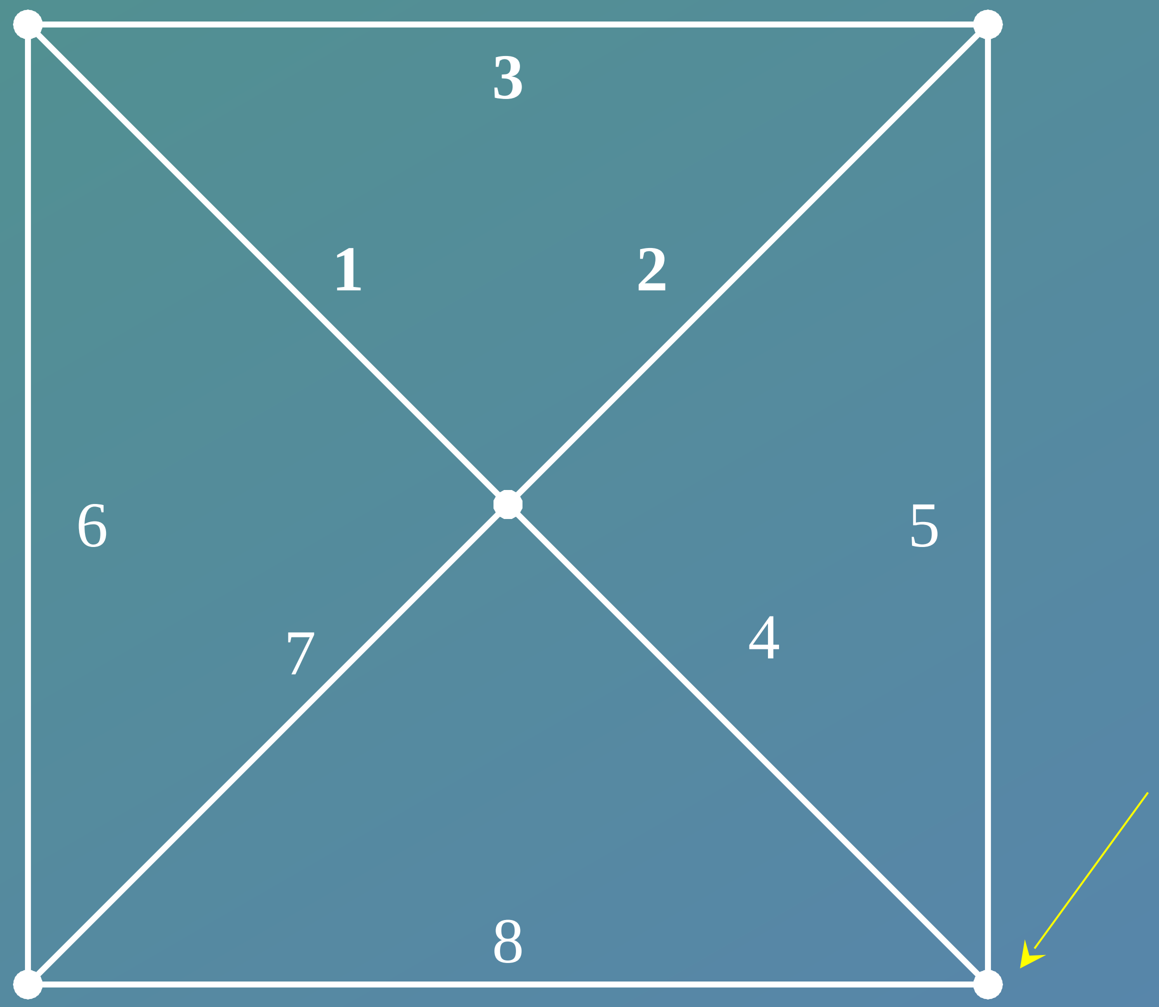
Construire la double séquence:

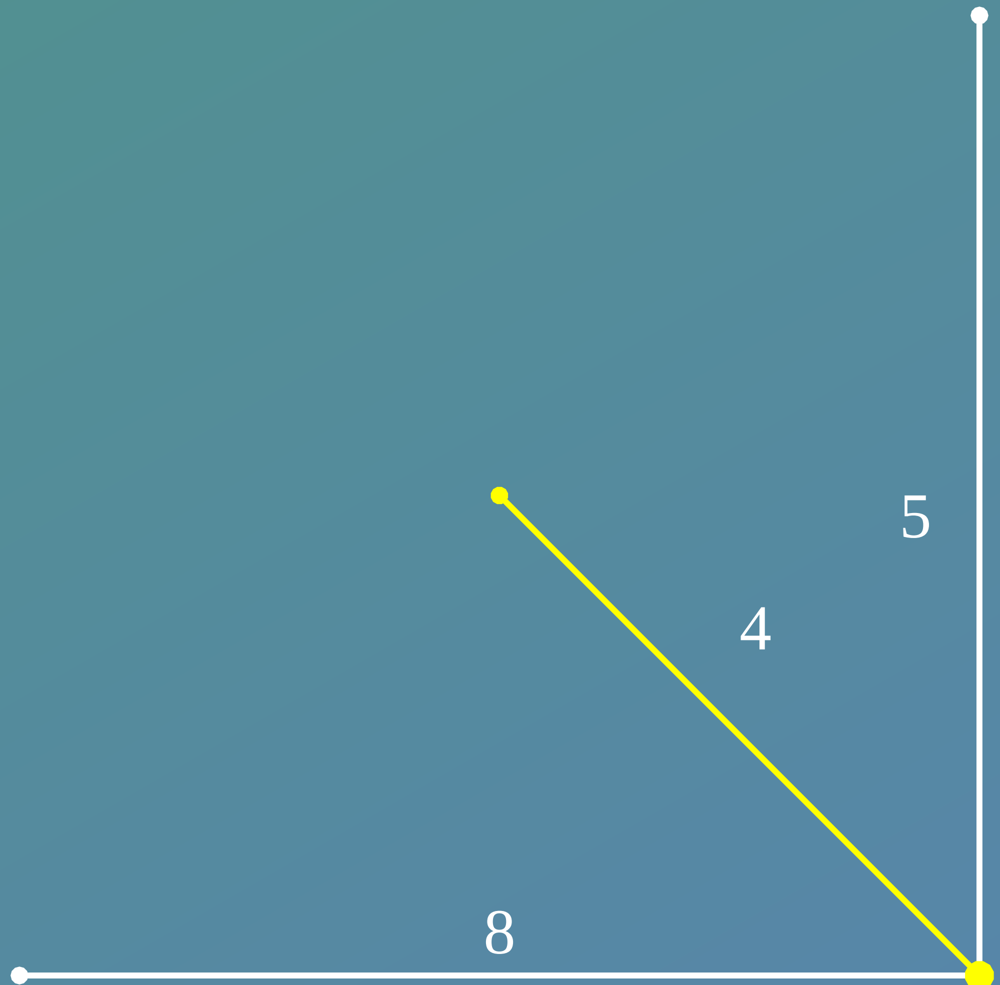
$\boldsymbol{T_1=\emptyset; \quad S_1=\{x_1\}}$

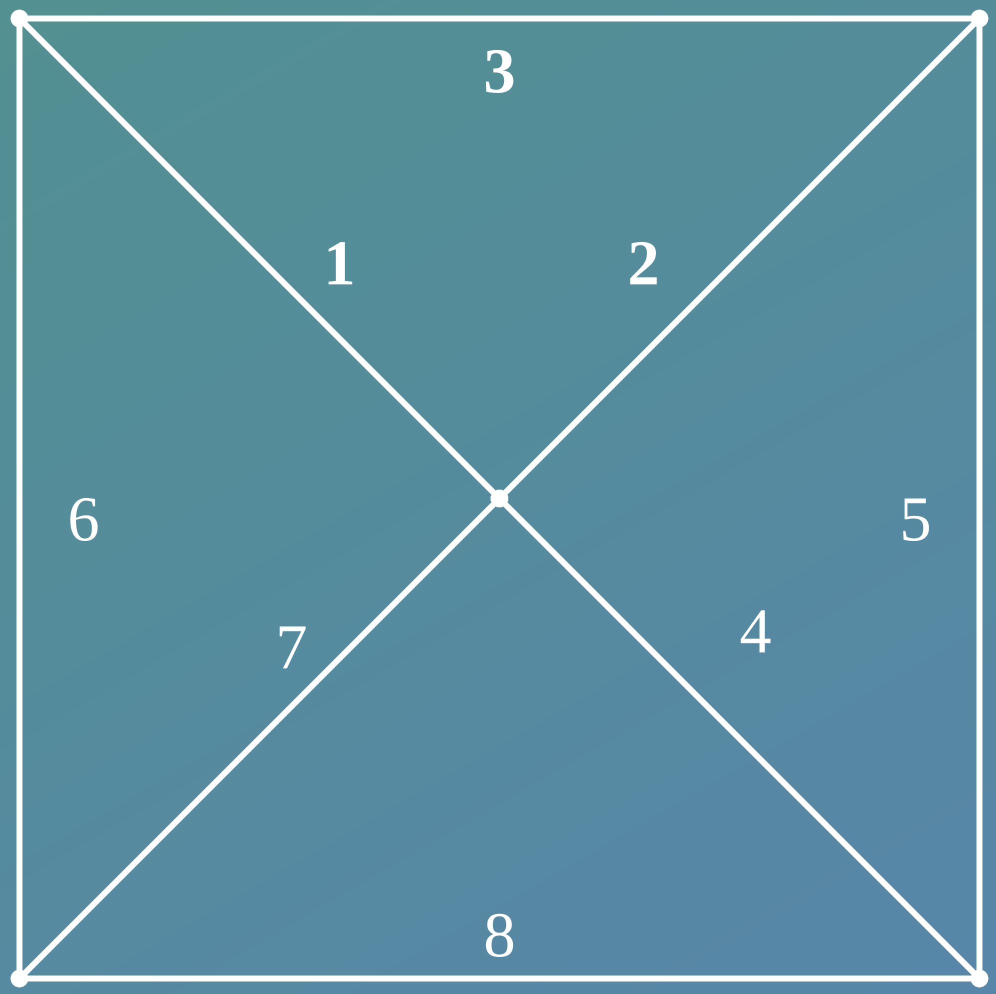
$\boldsymbol{T_{i+1}=T_i \cup \{e_k\}; \quad S_{i+1}=S_i \cup \{x_{i+1}\};}$

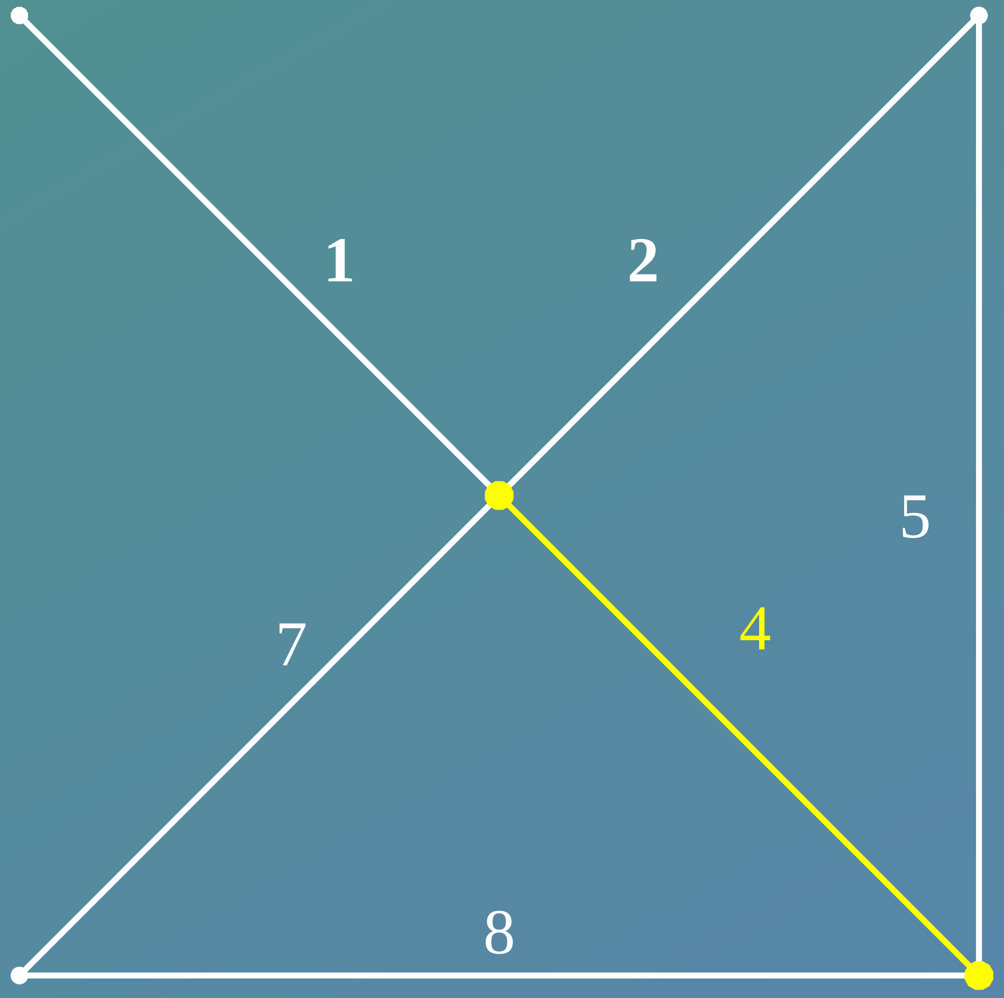
où $e = \{x, x_{i+1}\}$ *est une arête de poids minimum du cocycle* $\omega(S_i)$.

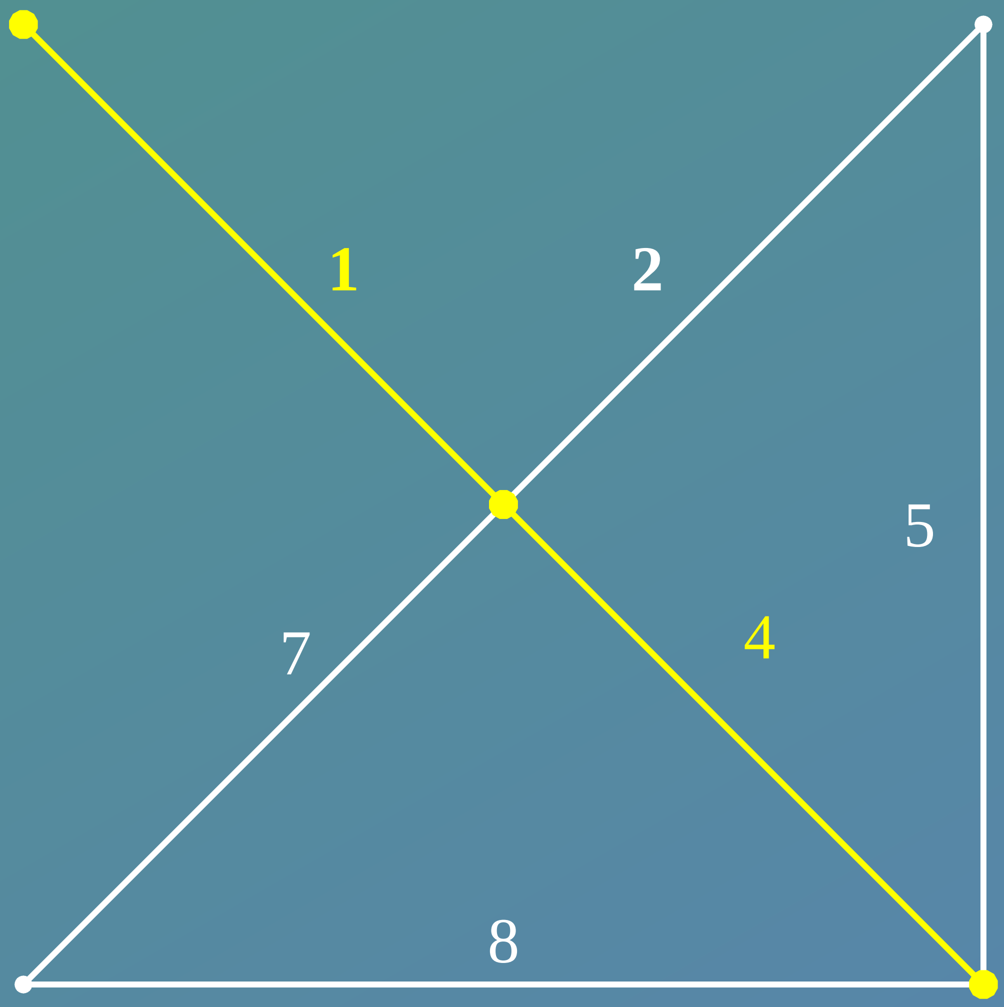
$\boldsymbol{A_n=(S_n;T_n)=(X;T_n)}$ *est un arbre de poids minimum.*

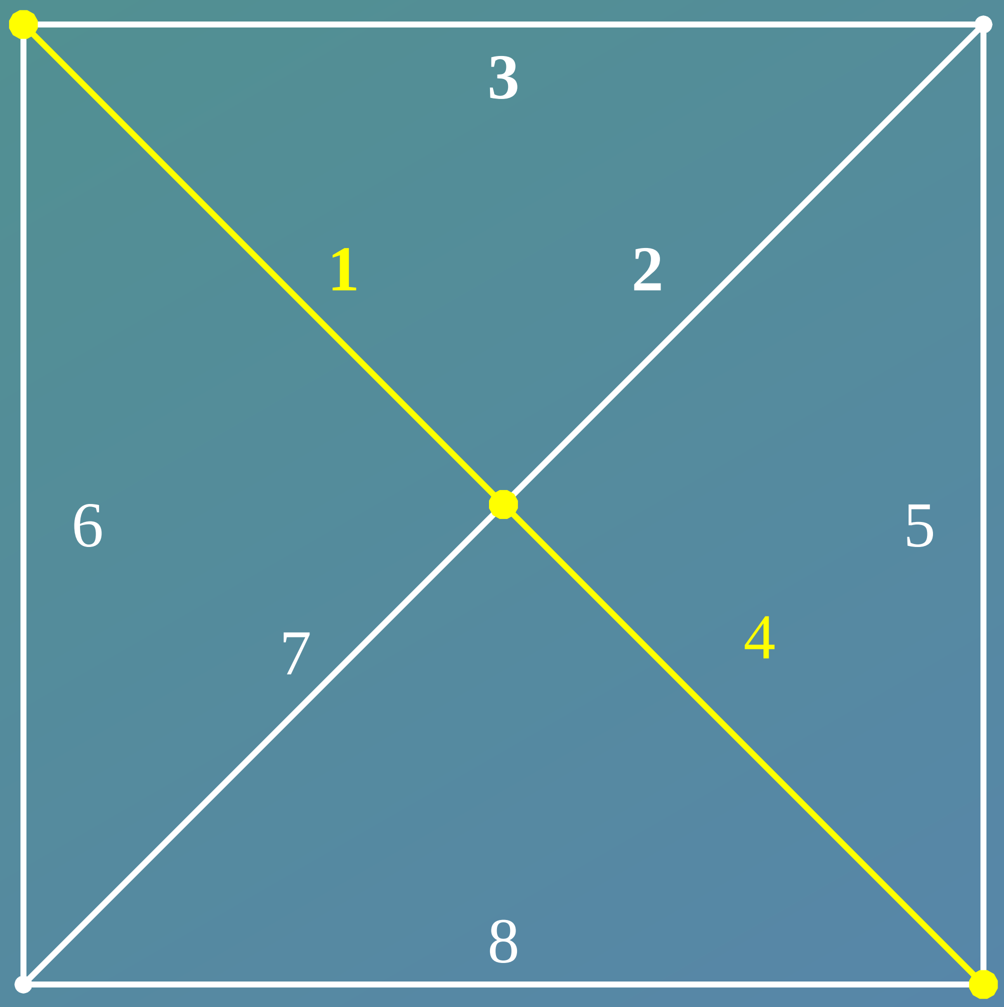

3
1
2
6
5
7
4
8

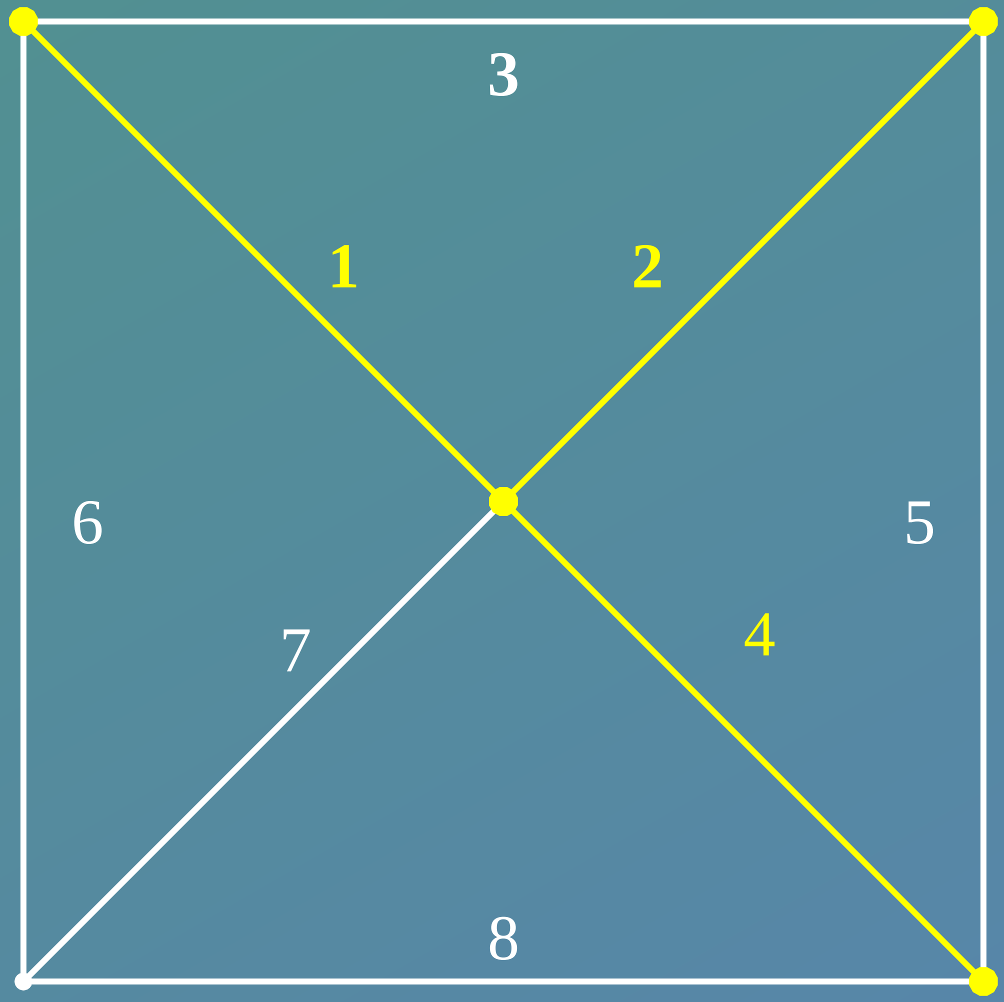

5
4
8

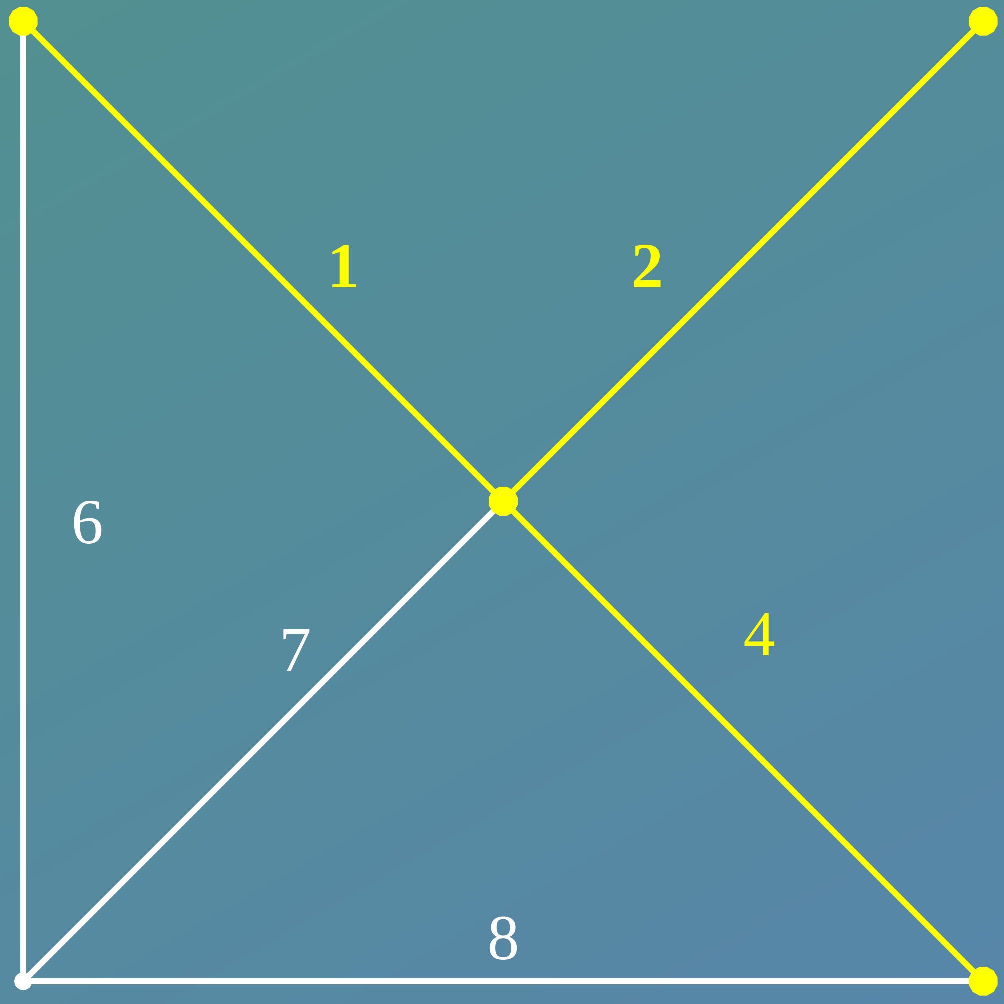

3
1
2
6
5
7
4
8


1
2
5
7
4
8

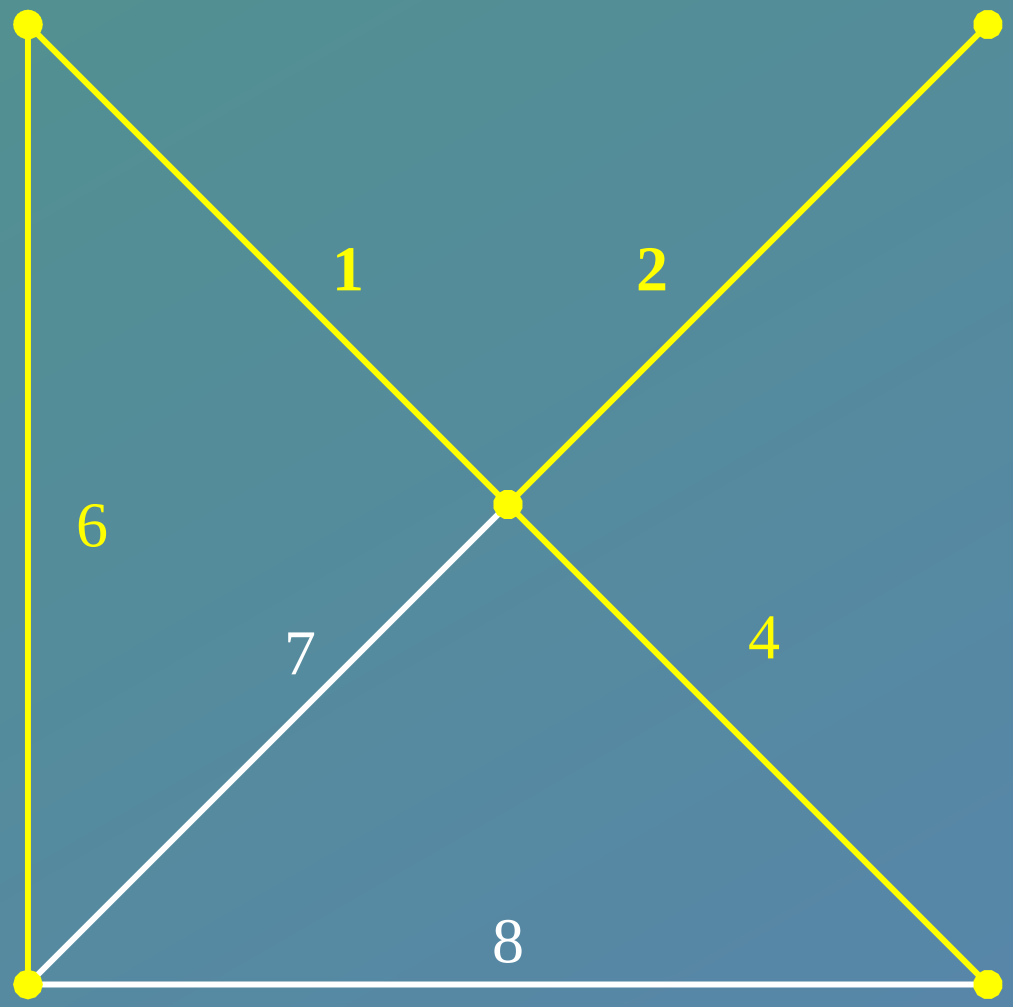

1
2
5
7
4
8


3
1
2
6
5
7
4
8

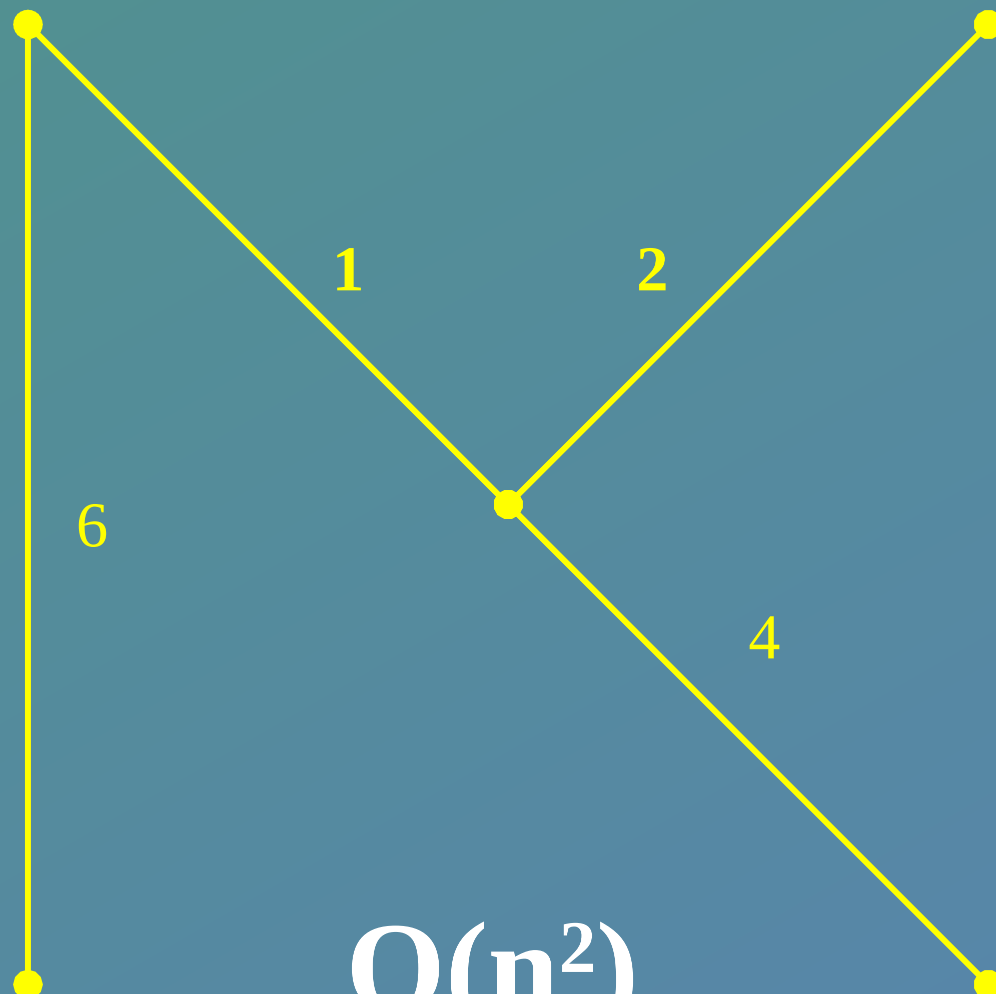

3
1
2
6
5
7
4
8



1
2
6
7
4
8



1
2
6
7
4
8



$O(n^2)$

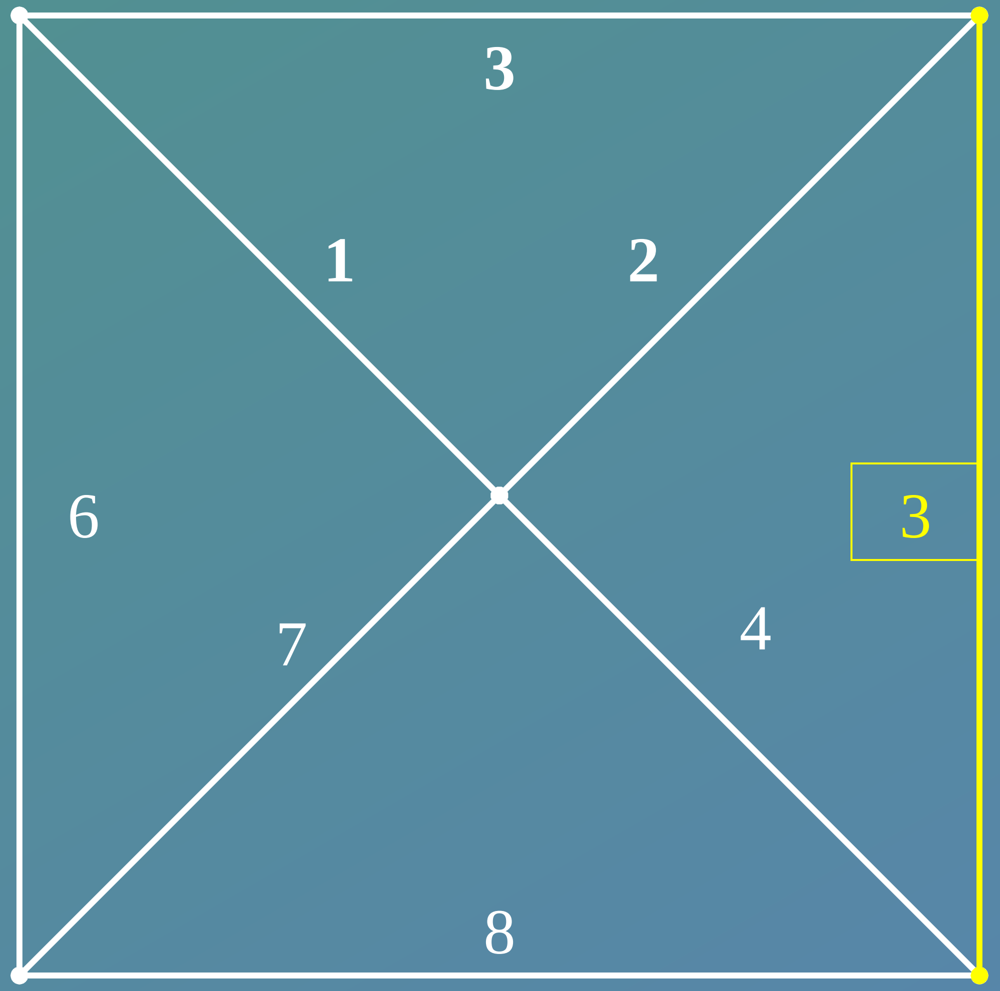
Algorithme de

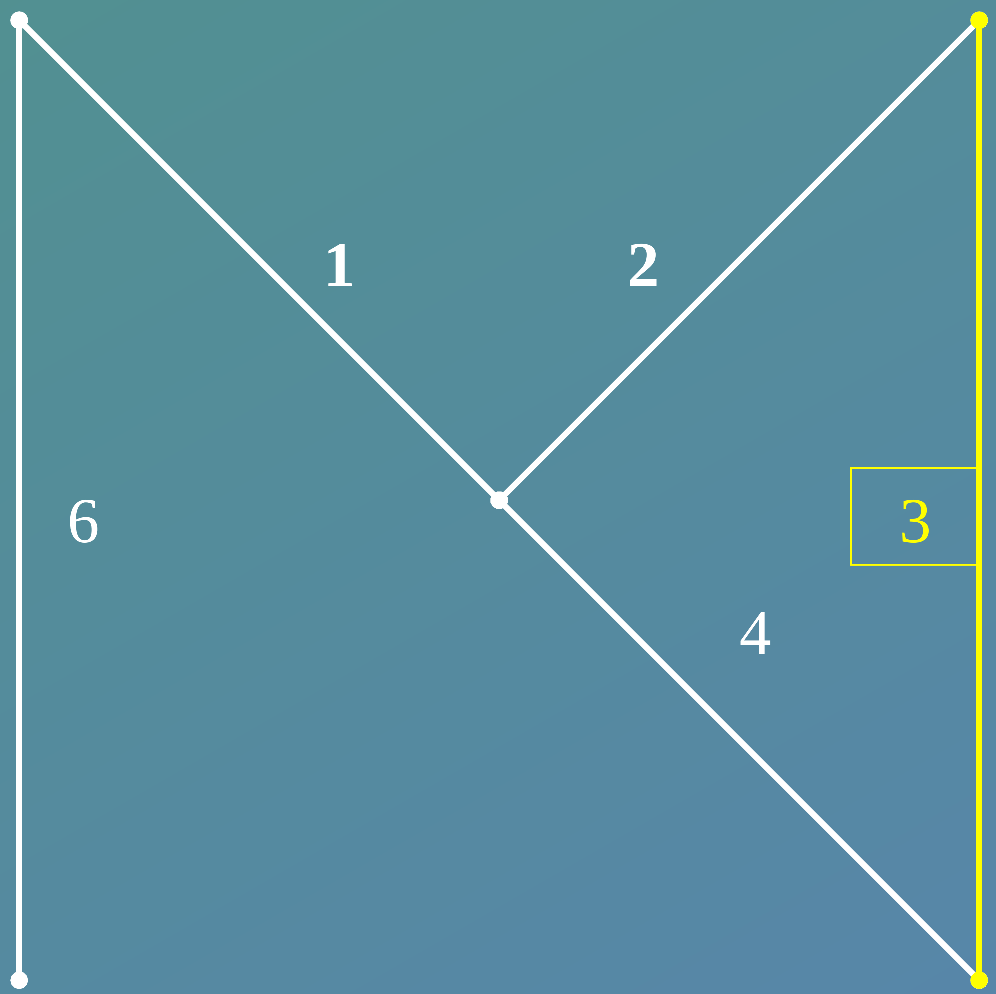
Kruskal:

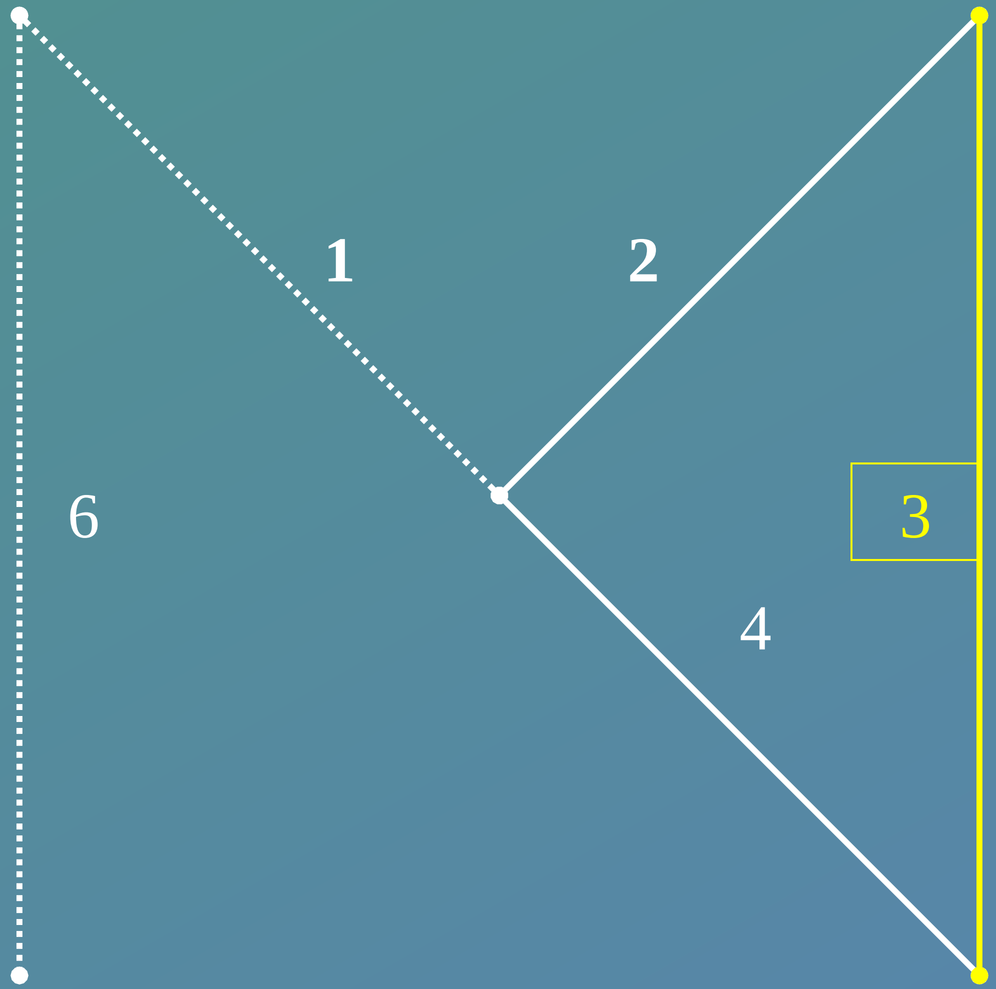
$O(mlgm)$

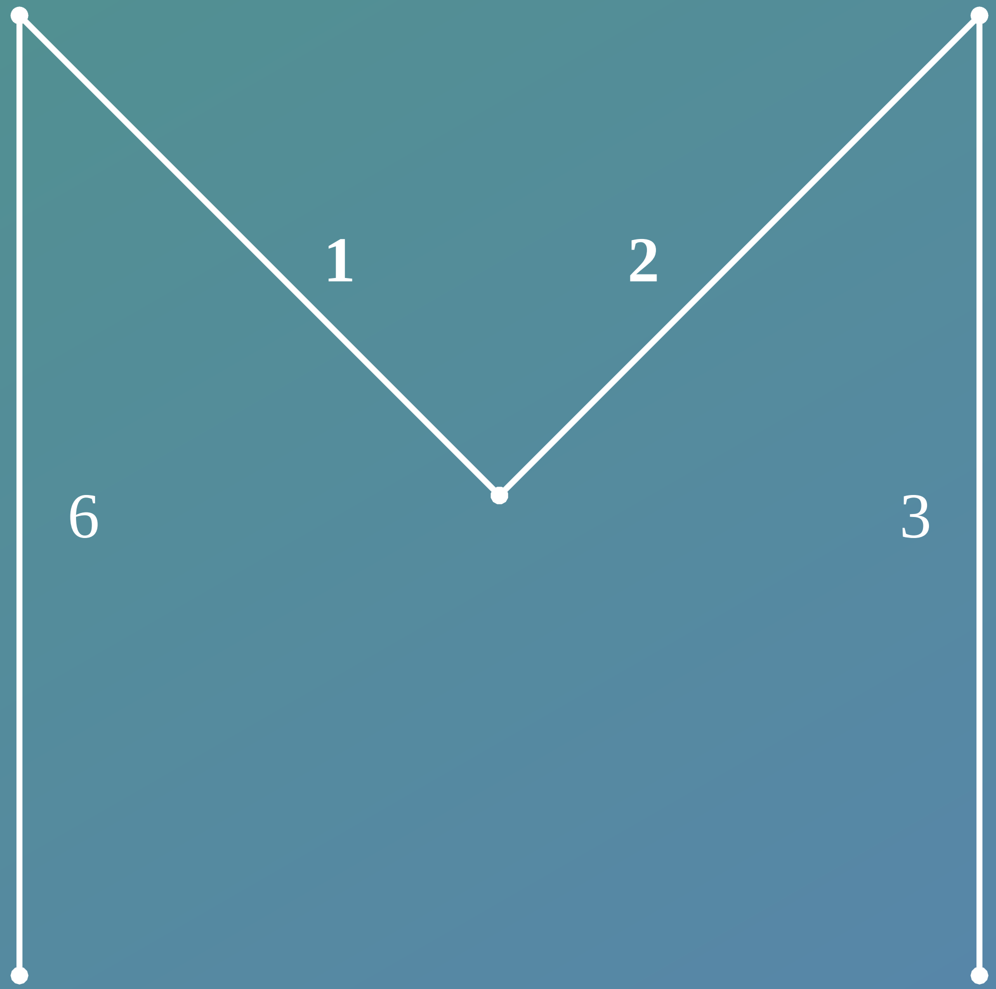
Prim:

$O(n^2)$


3
1
2
6
3
7
4
8


1
2
3
4
6

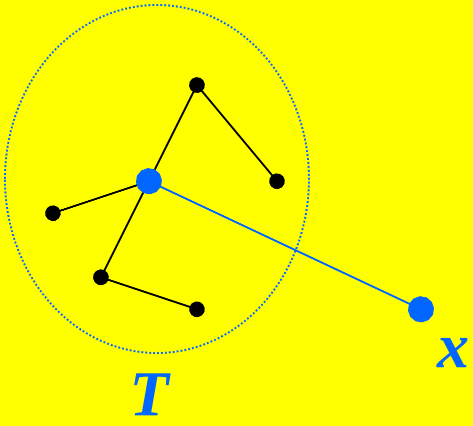

1
2
3
4
6


1
2
6
3

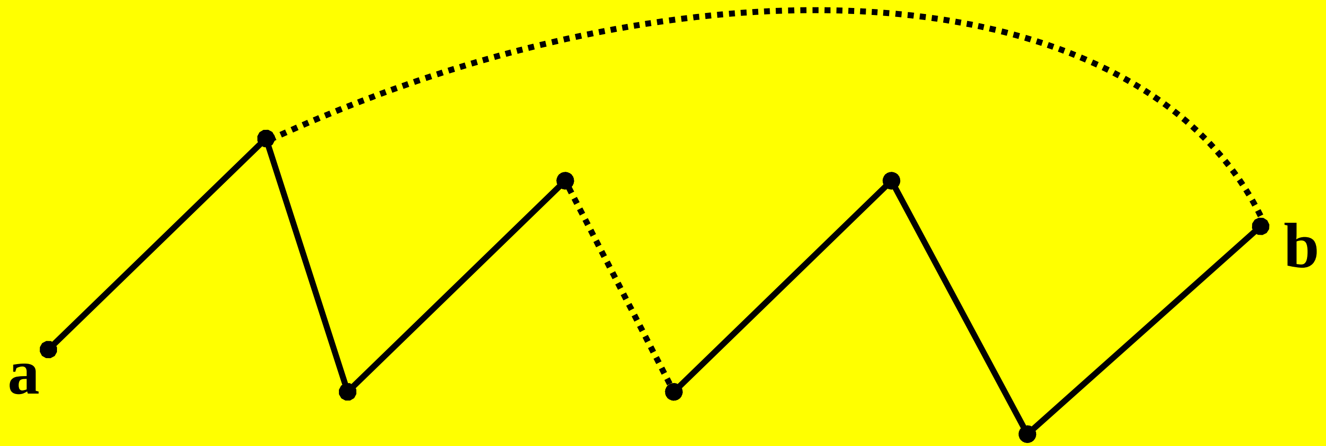
Théorème 2 :

*La classe **A** des arbres est définie inductivement par le schéma suivant:*

- *base: le graphe à un sommet;*
- *règle: Si* $T \in \mathbf{A}$ *alors* $T+x \in \mathbf{A}$.



Tout arbre, d'ordre au moins 2, possède au moins deux sommets pendants (des sommets de degré 1).



Théorème 5 :

Soit $G=(X,E)$ un graphe d'ordre $|X|= n \geq 2$. Les propriétés suivantes sont équivalentes:

(1) *G est **connexe** et **sans cycles**;*

(2) *G est **sans cycles** et admet **n–1** arêtes;*

(3) *G est **connexe** et admet **n–1** arêtes;*

Théorème 6 :

Les propriétés suivantes sont équivalentes:

(1) *G=(X,E) est* ***connexe*** *et* ***sans cycles****;*

(2) *G est* ***connexe-minimal*** *(si on supprime une arête quelconque, il n'est plus connexe);*

(3) *G est sans boucle et tout couple de sommets est relié par une chaîne unique.*

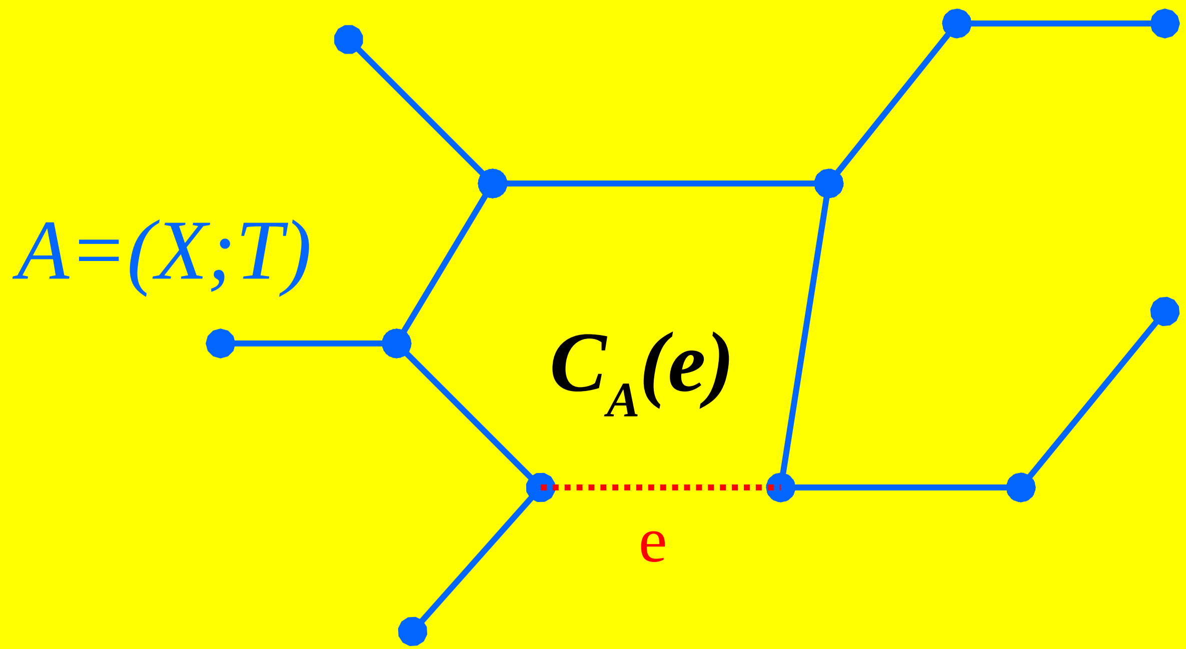
Théorème 7 :

Soit $G=(X,E)$ un graphe connexe, alors:

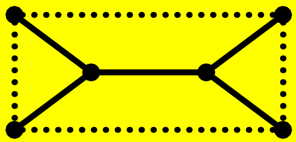
(1) *$A=(X;T)$ est un arbre de $G \Leftrightarrow A$ est* ***sans cycles*** *et maximal pour cette propriété; en ajoutant une arête e de $E\backslash T$, on crée un cycle unique dans $A+e$;*

(2) *$K=(X;F)$ est un co-arbre de $G \Leftrightarrow K$ ne contient pas de cocycles de G et maximal pour cette propriété; en ajoutant une arête e de $E\backslash F$, on crée un cocycle unique dans $A+e$;*

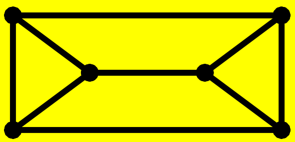
(1) *A=(X;T) est un arbre de G* $\Leftrightarrow$ *A est* ***sans cycles*** *et maximal pour cette propriété; en ajoutant une arête e de E\T, on crée un cycle unique dans A+ e;*



(2) *K=(X;F) est un co-arbre de G* **⇔** *K ne contient pas de cocycles de G et maximal pour cette propriété; en ajoutant une arête e de E\F, on crée un cocycle unique dans A+ e;*

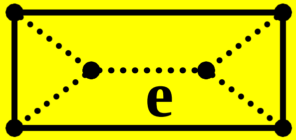


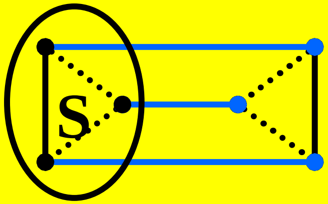
un arbre couvrant A



graphe G

co-arbre associé K





$\boldsymbol{\omega_K(e)} = \omega_G(S)$

Théorème 9:

Soit $A=(X;T)$ un arbre du graphe connexe $G=(X,E)$ et $K=(X;E\setminus T)$ le co-arbre associé à A et $p: E \rightarrow R$ une fonction qui associe à toute arête de G un poids réel.

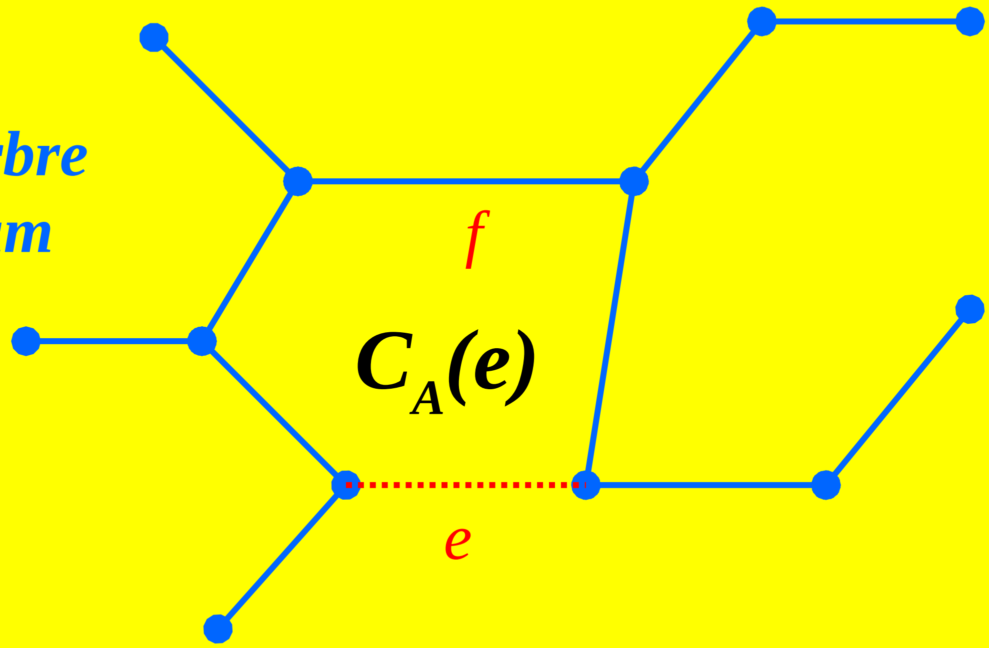
Les propriétés suivantes sont équivalentes:

(1) $A=(X;T)$ est un arbre de poids minimum;

(2) $\forall e \in E\setminus T$, $\forall f \in C_A(e)$ on a: $p(f) \leq p(e)$;

(3) $\forall e \in T$, $\forall f \in \omega_K(e)$ on a: $p(f) \geq p(e)$.

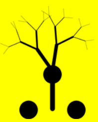
A=(X;T) est un arbre de poids minimum



(2) $\forall e \in E \backslash T,$

$\forall f \in C_A(e)$

on a: $p(f) \leq p(e);$



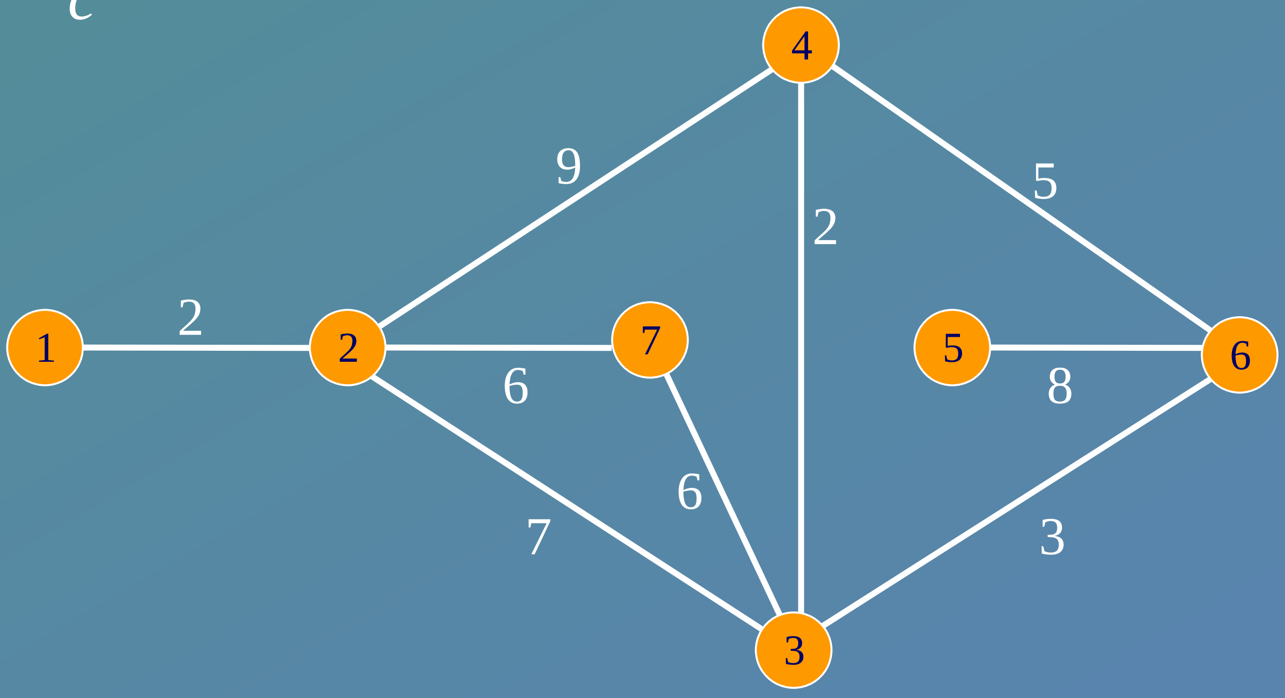
Exercice 1:

Un réseau informatique est composé de sept stations de travail localisées en sept centres différents. Certains centres sont reliés entre eux par des lignes de communication. Les équipements étant vétustes, il a été décidé de procéder à leur remplacement par un matériel plus moderne. Ce remplacement peut être effectué selon les coûts donnés par la matrice suivante :

	2	3	4	5	6	7
1	2	*c*	*c*	*c*	*c*	*c*
2		7	9	*c*	*c*	6
3			2	*c*	3	6
4				*c*	5	*c*
5					8	*c*
6						*c*

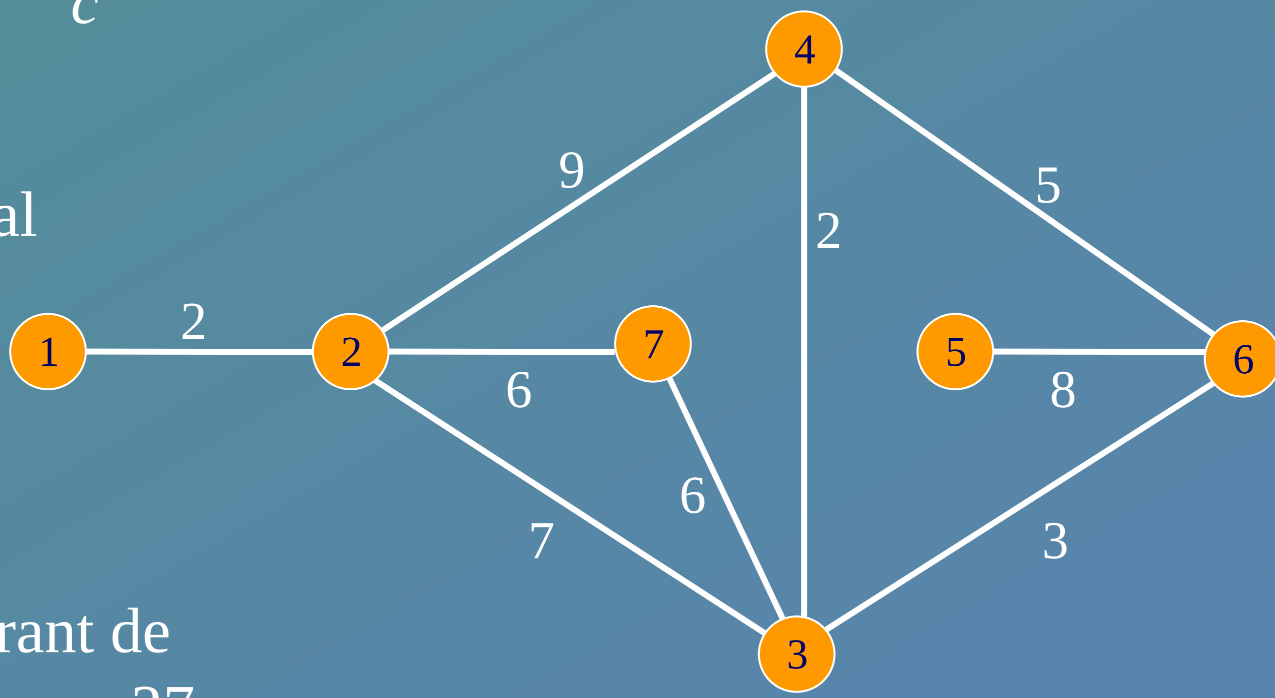
La valeur *c* veut dire qu'il n'existe pas de ligne reliant directement les deux centres correspondants, mais on peut la construire pour un coût $c = 10$.

	2	3	4	5	6	7
1	2	*c*	*c*	*c*	*c*	*c*
2		7	9	*c*	*c*	6
3			2	*c*	3	6
4				*c*	5	*c*
5					8	*c*
6						*c*



	2	3	4	5	6	7
1	2	*c*	*c*	*c*	*c*	*c*
2		7	9	*c*	*c*	6
3			2	*c*	3	6
4				*c*	5	*c*
5					8	*c*
6						*c*

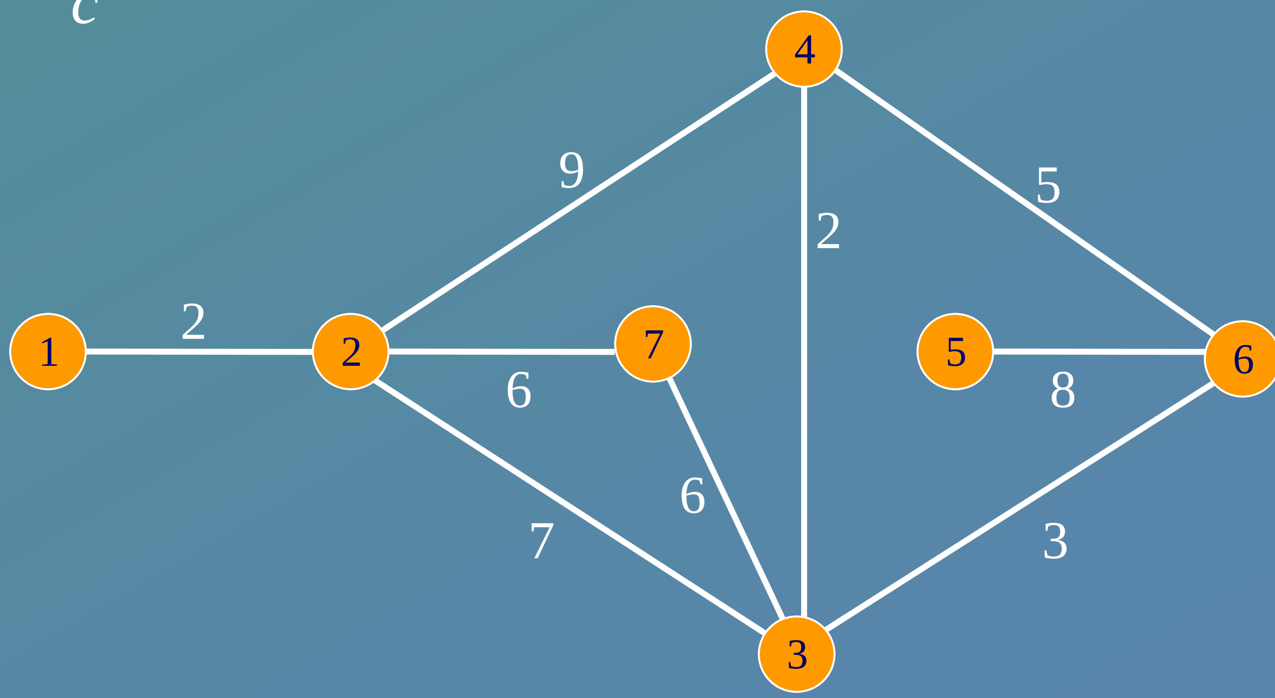
Algorithme de Kruskal



l’arbre couvrant de
poids minimum = 27

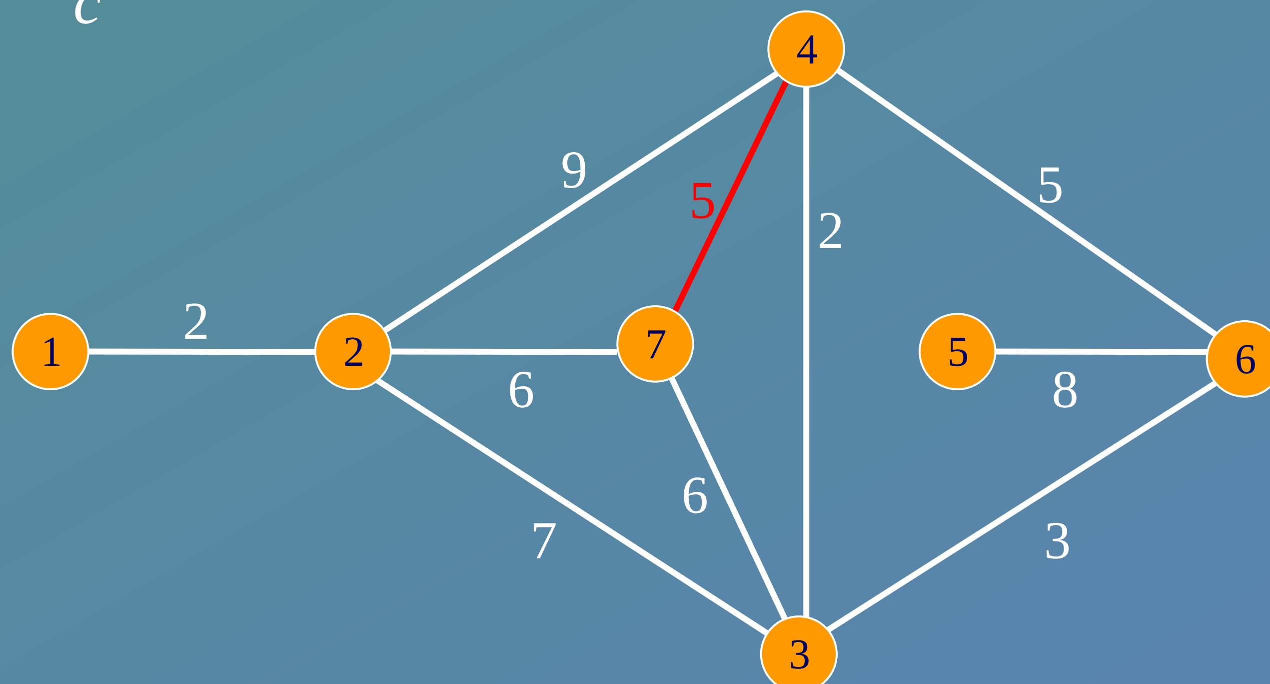
	2	3	4	5	6	7
1	2	c	c	c	c	c
2		7	9	c	c	6
3			2	c	3	6
4				c	5	c
5					8	c
6						c

Que peut-on dire si le coût C de construction de la ligne (4;7) diminue et devient 5 ?

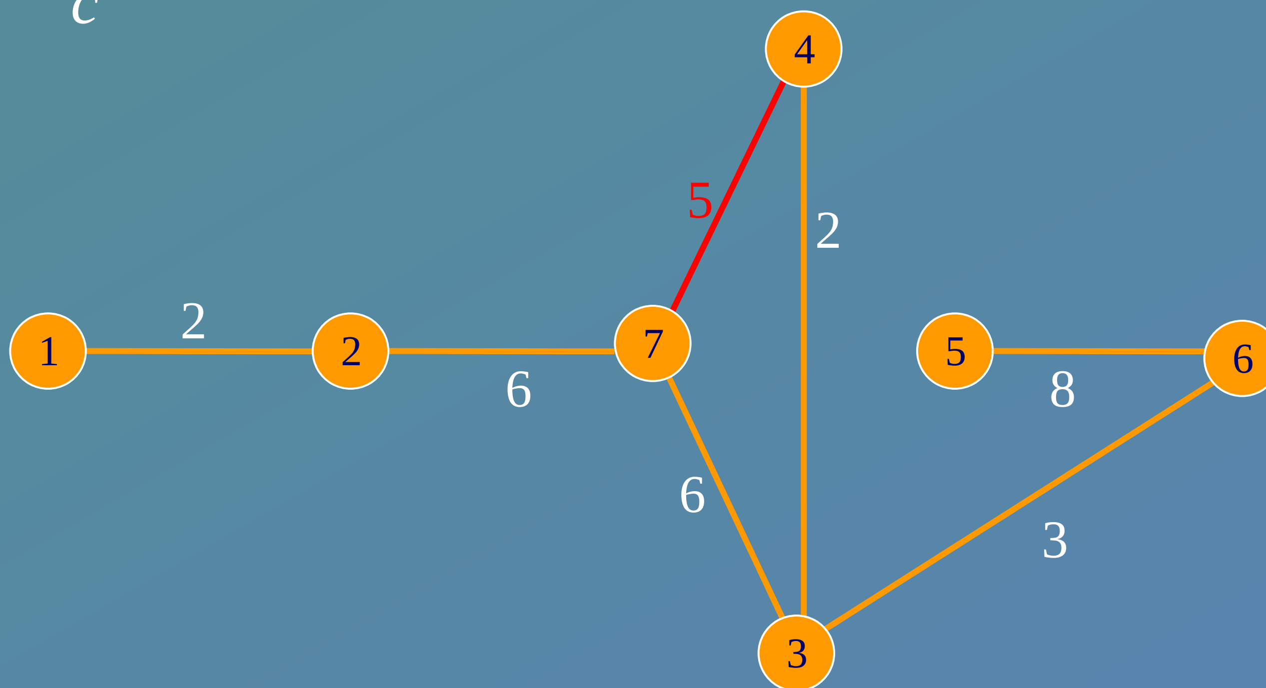


	2	3	4	5	6	7
1	2	c	c	c	c	c
2		7	9	c	c	6
3			2	c	3	6
4				c	5	5
5					8	c
6						c

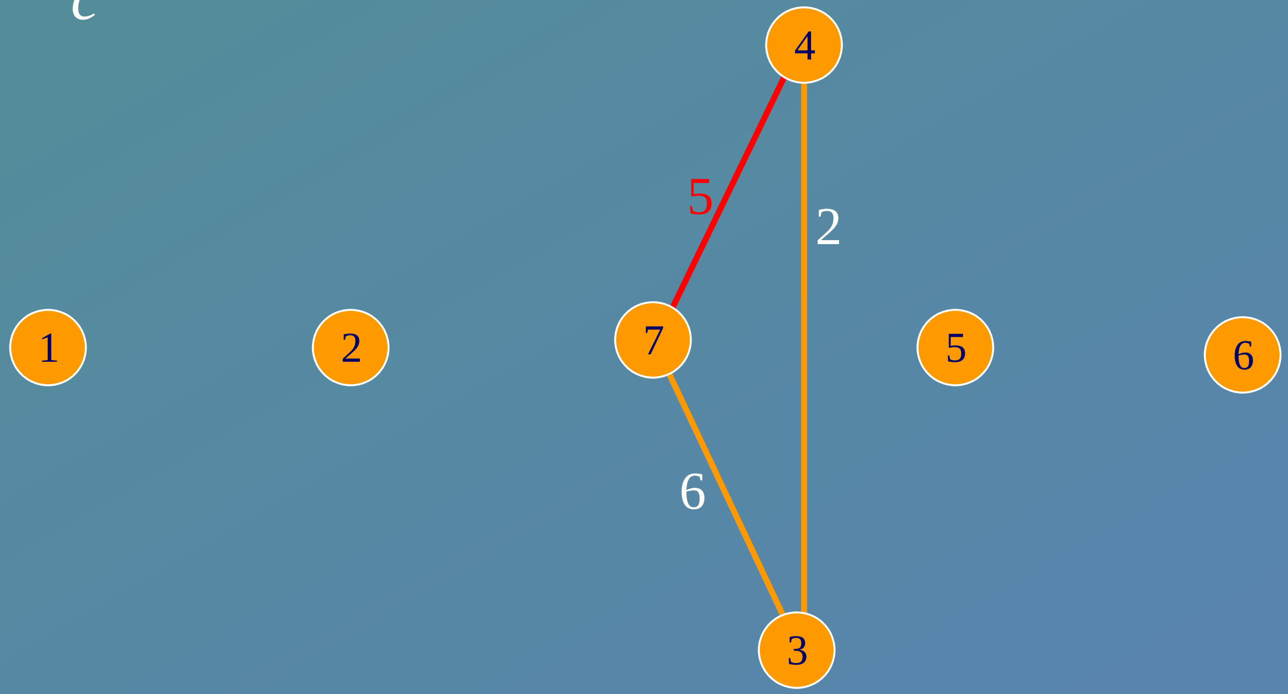
Que peut-on dire si le coût C de construction de la ligne (4;7) diminue et devient 5 ?



	2	3	4	5	6	7
1	2	c	c	c	c	c
2		7	9	c	c	6
3			2	c	3	6
4				c	5	5
5					8	c
6						c

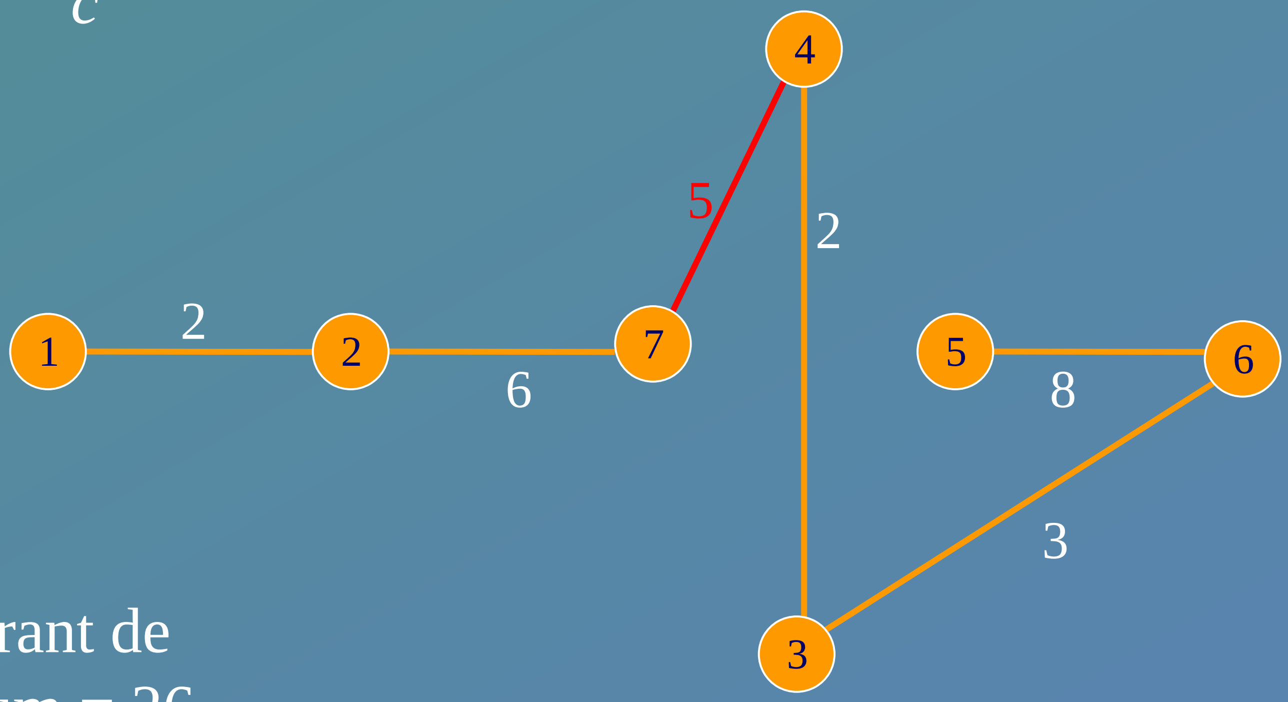


	2	3	4	5	6	7
1	2	*c*	*c*	*c*	*c*	*c*
2		7	9	*c*	*c*	6
3			2	*c*	3	6
4				*c*	5	5
5					8	*c*
6						*c*



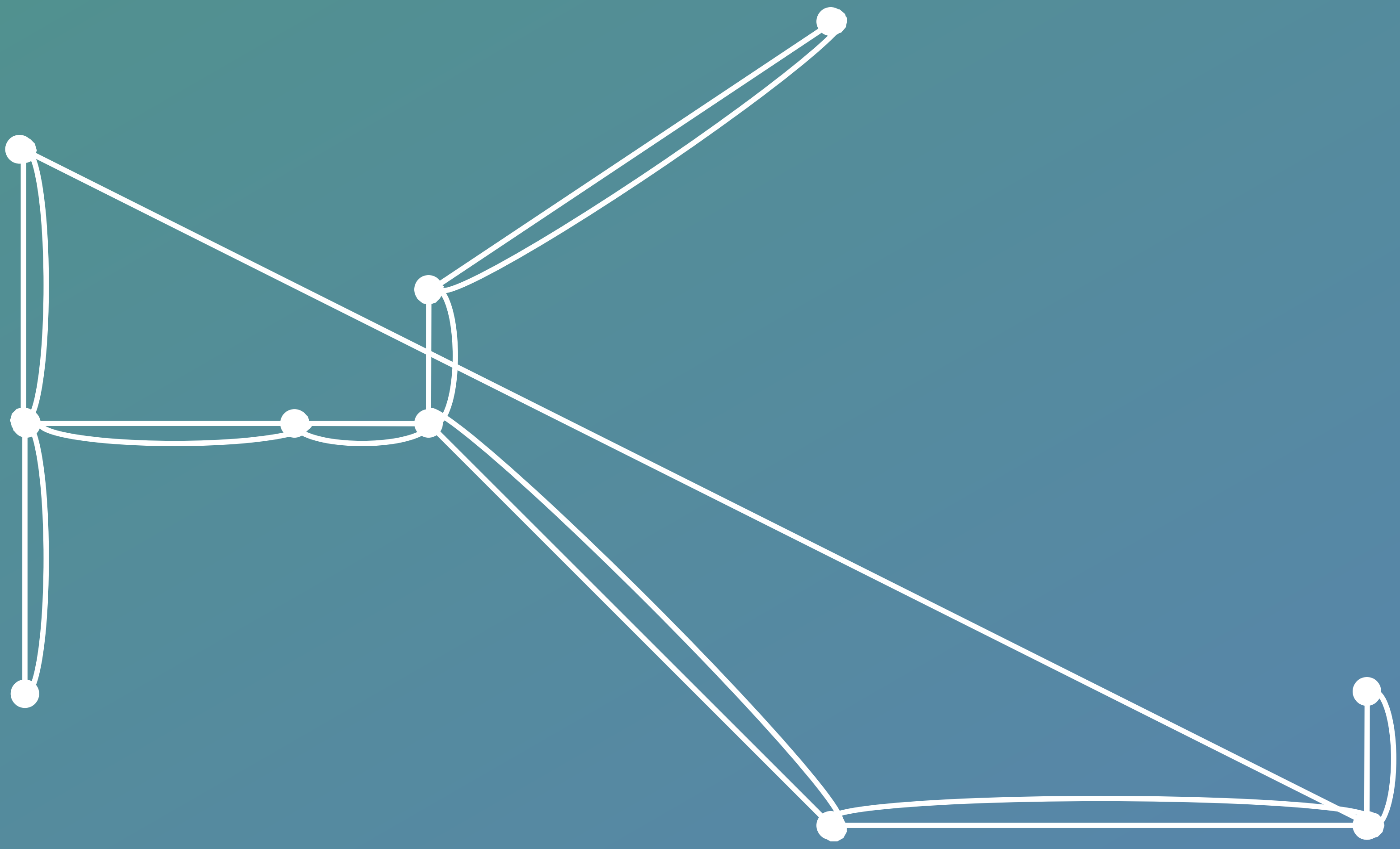
$+5 - 6 = -1$

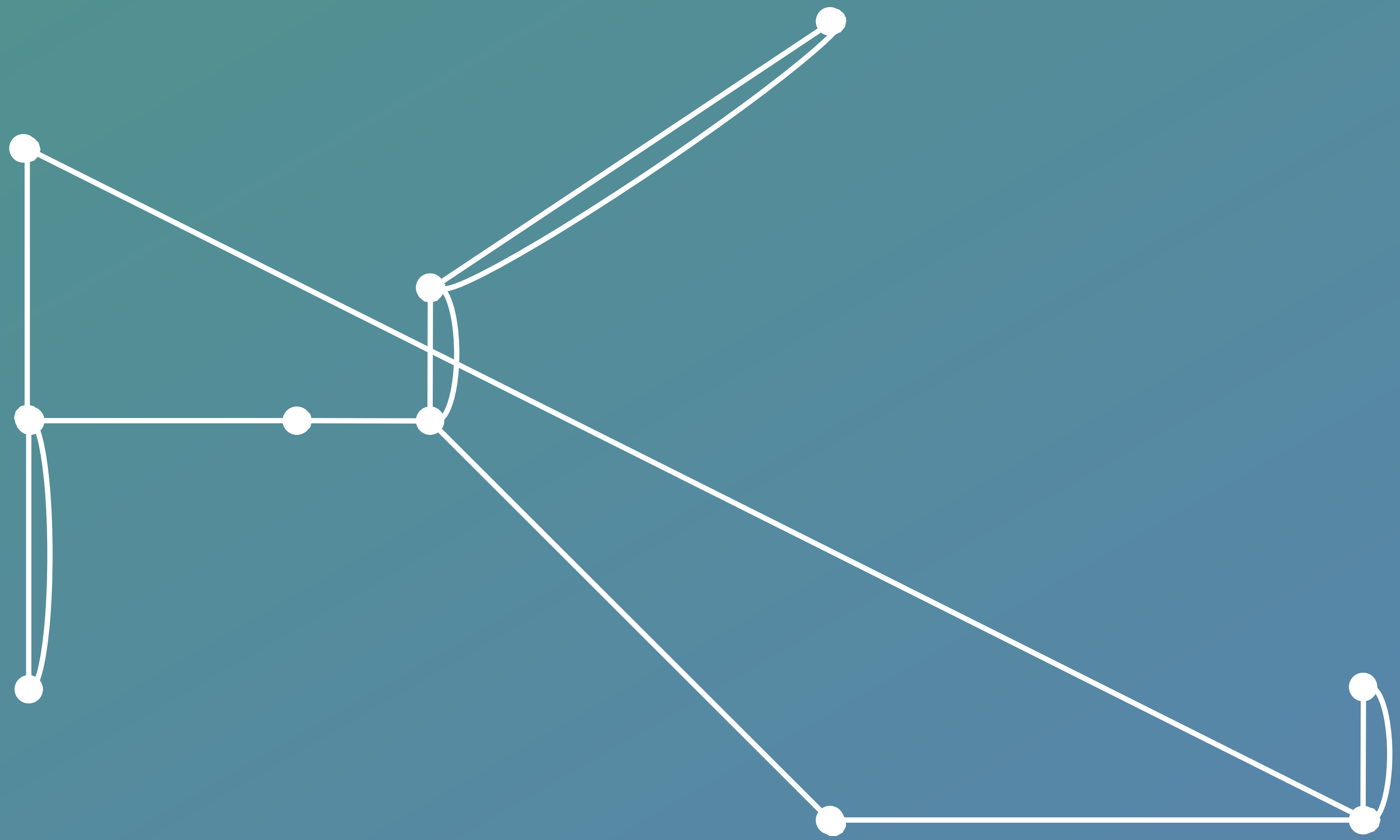
	2	3	4	5	6	7
1	2	*c*	*c*	*c*	*c*	*c*
2		7	9	*c*	*c*	6
3			2	*c*	3	6
4				*c*	5	5
5					8	*c*
6						*c*

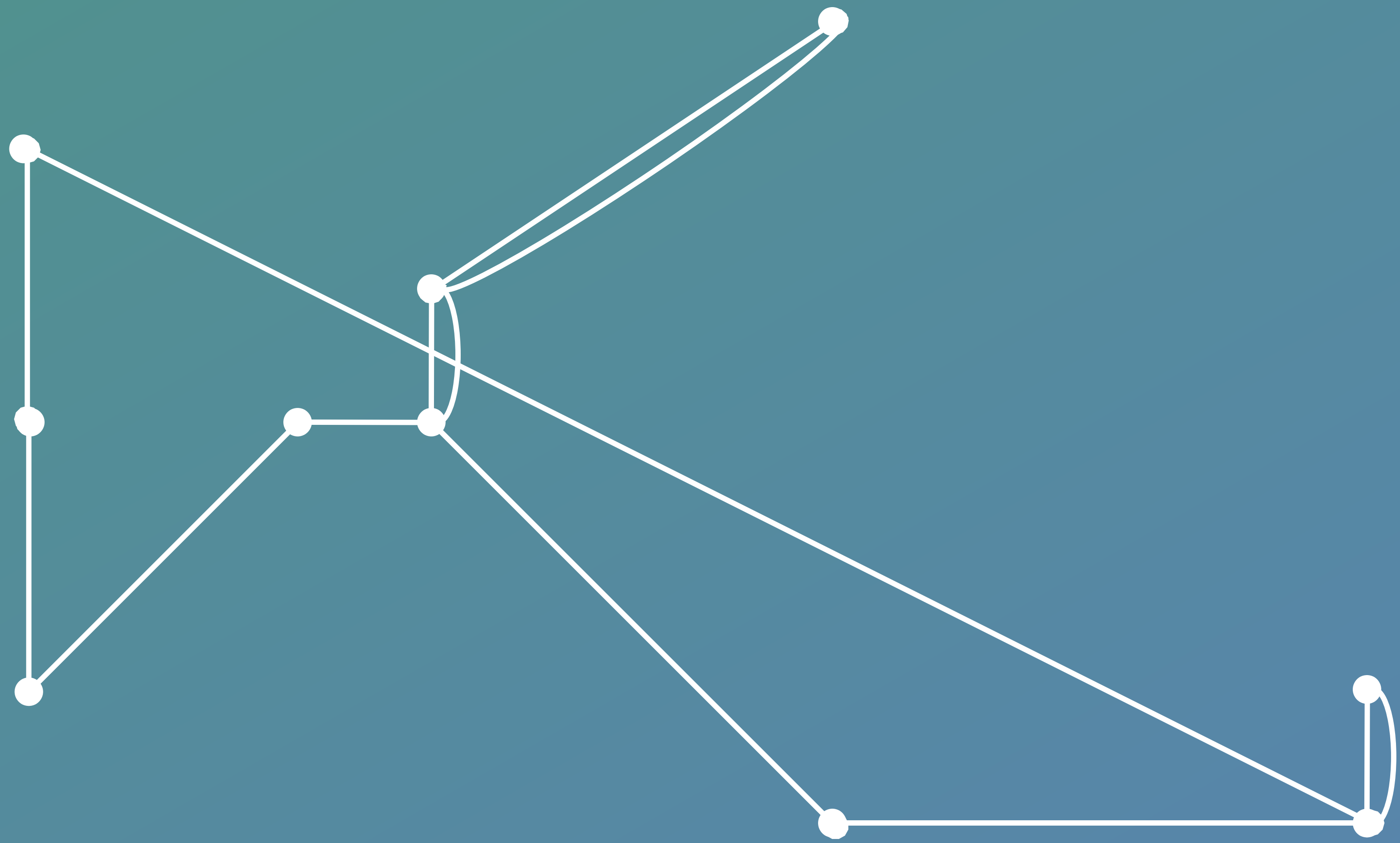


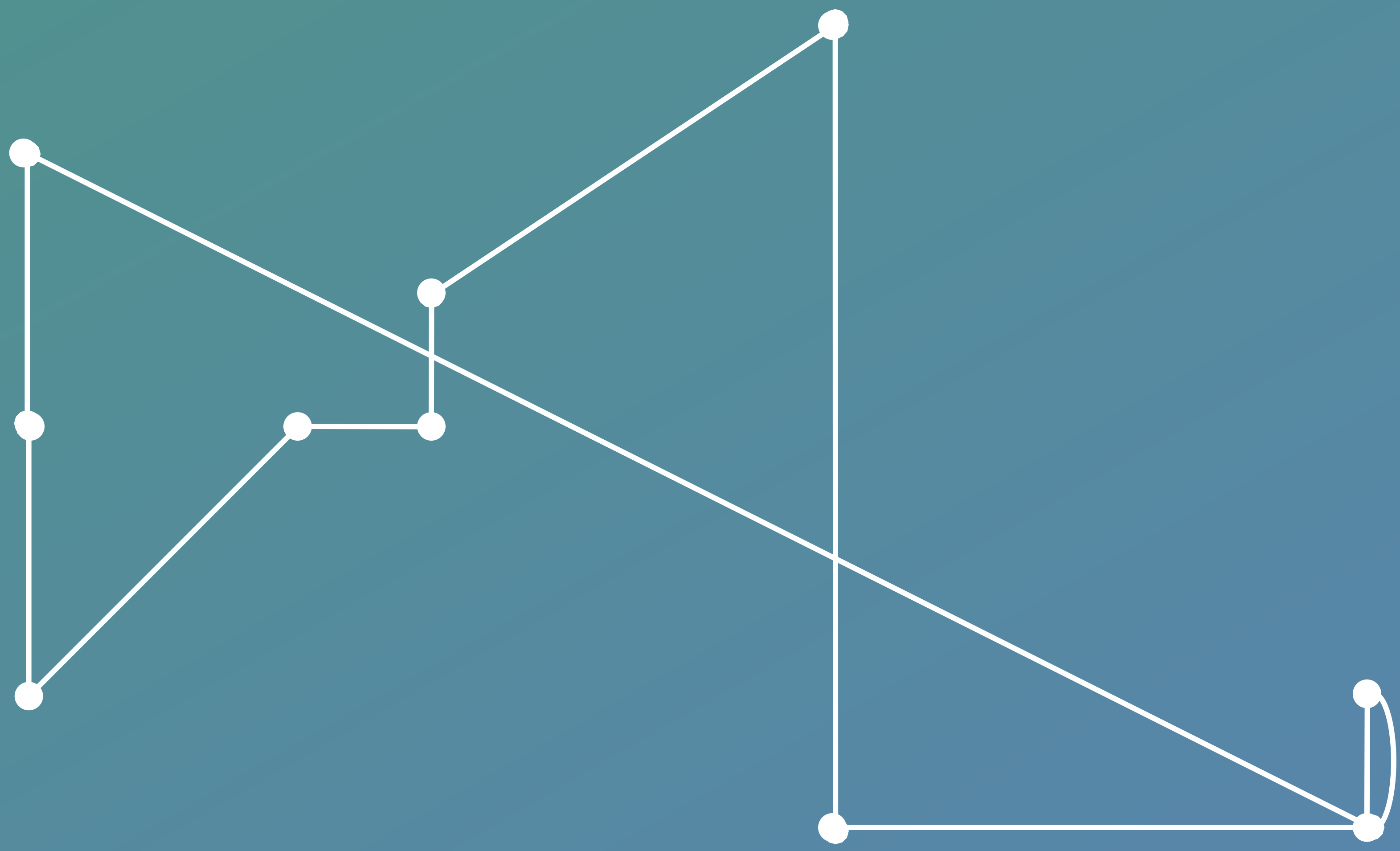
l'arbre couvrant de
poids minimum = 26

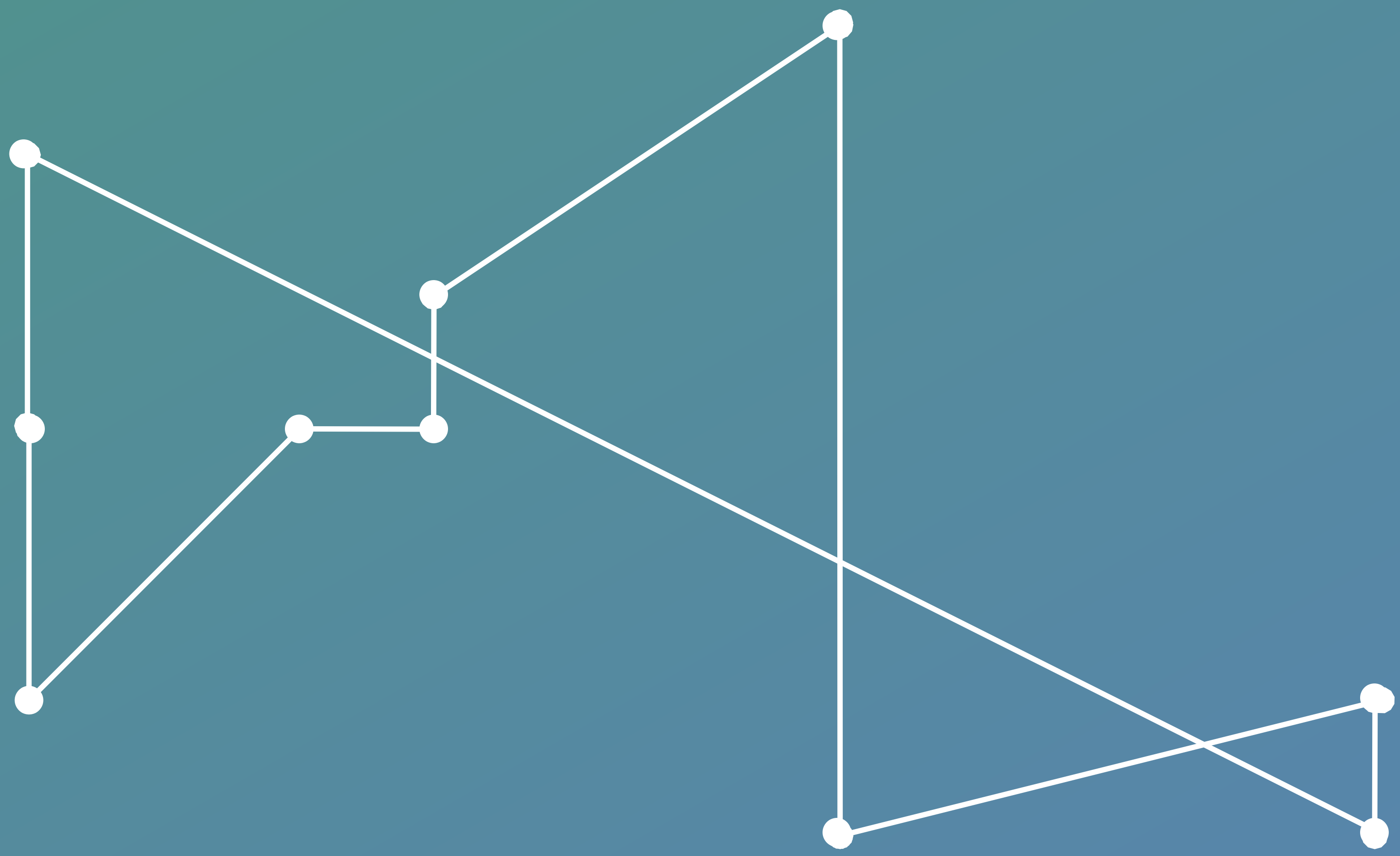
Algorithmes approximatifs pour le problème du voyageur de commerce : 1° Arbre de poids minimum

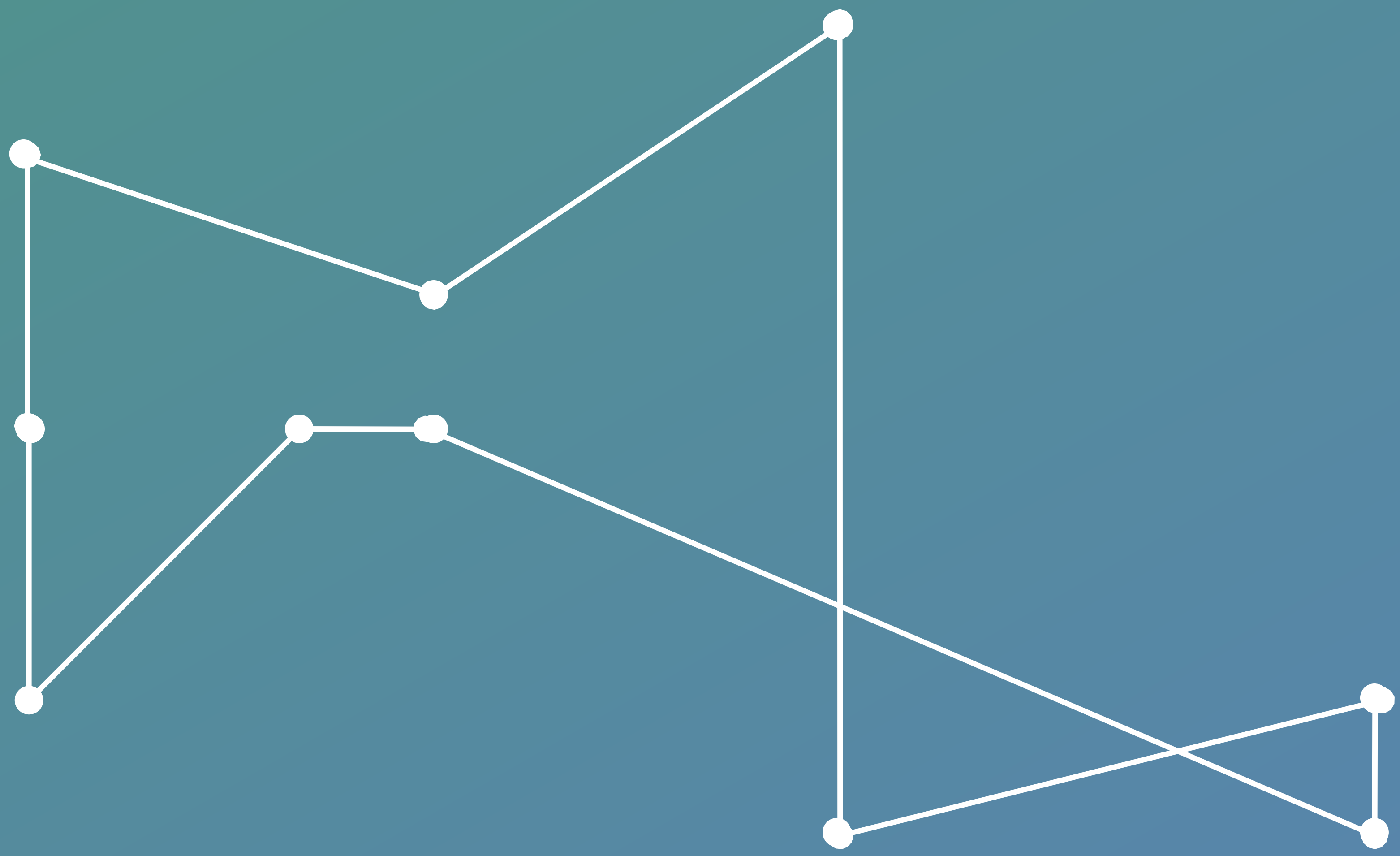


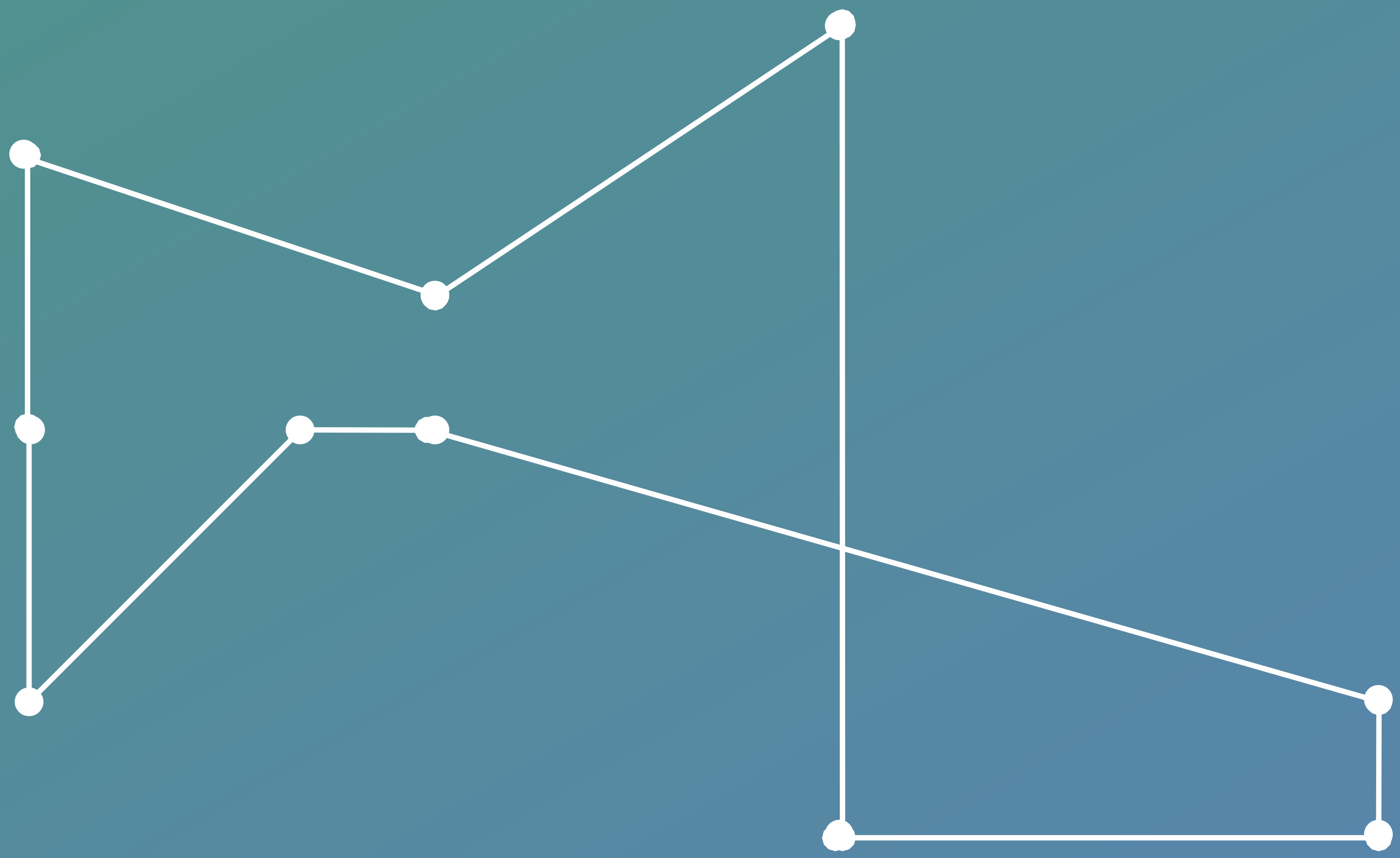












Algorithmes approximatifs pour le problème du voyageur de commerce : 2° Christofides