

# Algorithmique des graphes

## Cours 6 – Arbres couvrants de poids minimum

František Kardoš

`frantisek.kardos@u-bordeaux.fr`

# Arbres couvrants de poids minimum

Entrée : Un graphe (non-orienté)  $G$  connexe, avec une pondération des arêtes

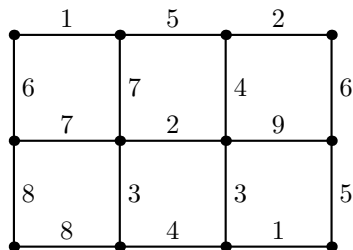
$$w : E(G) \rightarrow \mathbb{R}^+$$

Sortie : Un arbre couvrant  $T$  de  $G$ , de poids minimum

$$w(T) := \sum_{e \in E(T)} w(e)$$

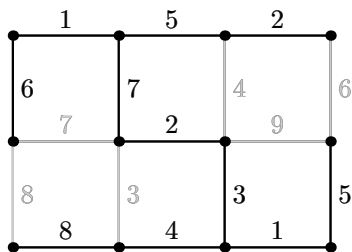
# Arbres couvrants de poids minimum

Entrée : Graphe connexe  $G$  avec les arêtes pondérées



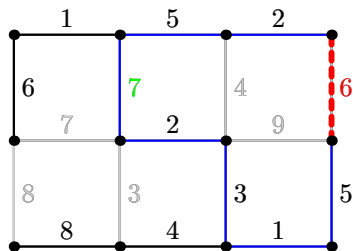
# Arbres couvrants de poids minimum

Sortie : Un arbre couvrant  $T$  de  $G$  :  $w(T) = 44$ .



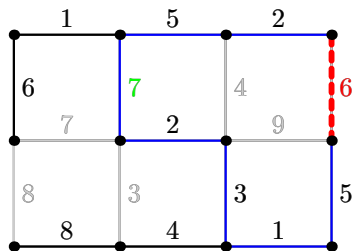
# Critère d'optimalité

Si dans une solution optimale  $T^*$  une arête n'est pas sélectionnée, toutes les arêtes du cycle qu'elle complète sont moins chères.



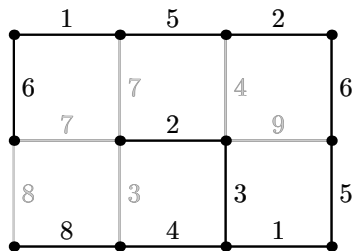
# Critère d'optimalité

Dans le cas contraire, on peut améliorer encore la solution actuelle en échangeant une arête utilisée trop chère pour une arête non-utilisée moins chère.



# Critère d'optimalité

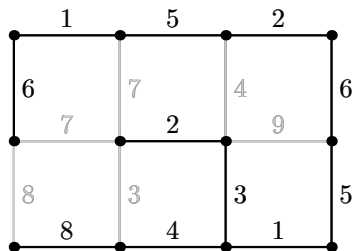
Dans le cas contraire, on peut améliorer encore la solution actuelle en échangeant une arête utilisée trop chère pour une arête non-utilisée moins chère.



# Critère d'optimalité

On continue ainsi de suite jusqu'à ce qu'il n'y ait plus d'amélioration possible.

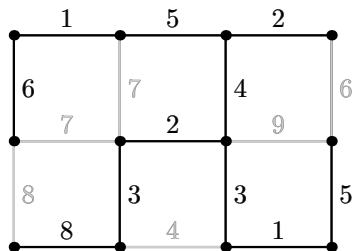
Combien d'améliorations encore possibles ?





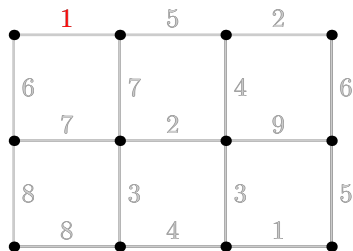
# Critère d'optimalité

Si dans une solution optimale  $T^*$  une arête n'est pas sélectionnée, toutes les arêtes du cycle qu'elle complète sont moins chères. Solution optimale :



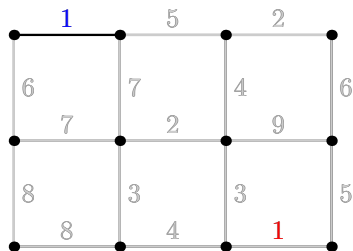
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



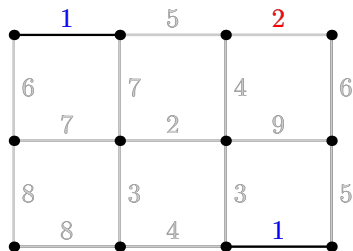
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



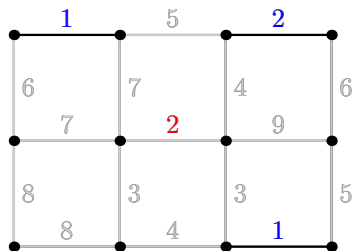
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



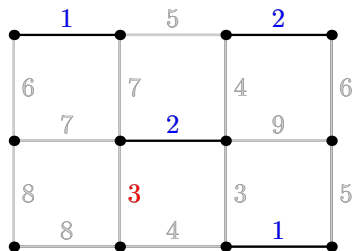
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



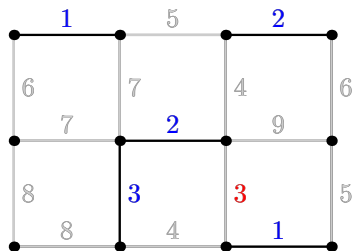
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



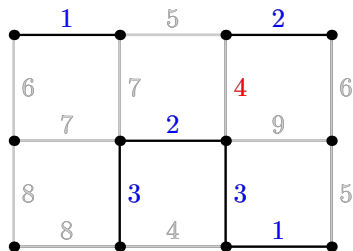
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



# Algorithme de Kruskal

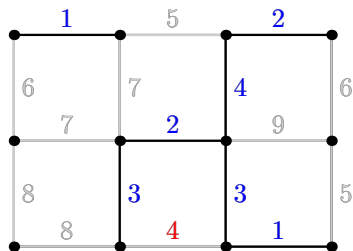
Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.





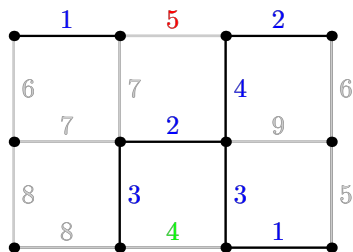
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



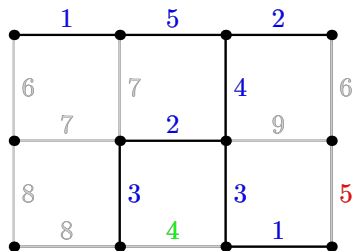
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



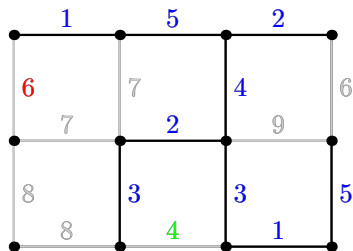
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



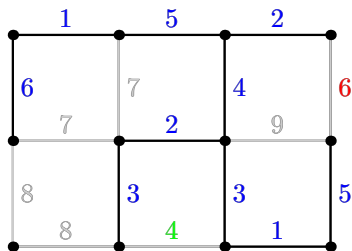
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



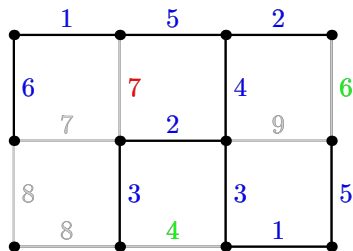
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



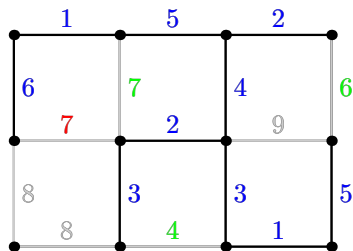
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



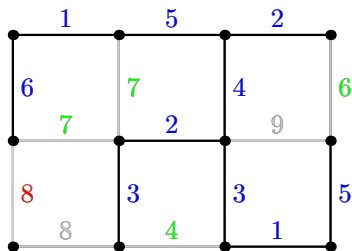
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



# Algorithme de Kruskal

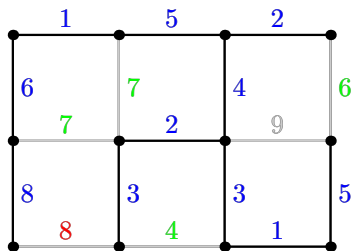
Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.





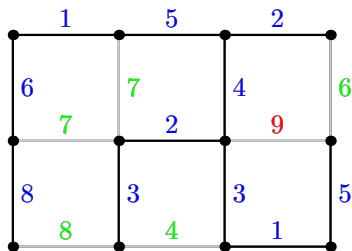
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



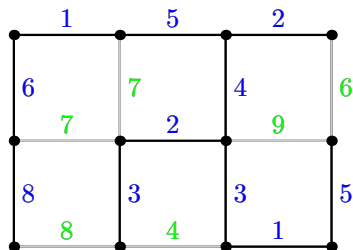
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



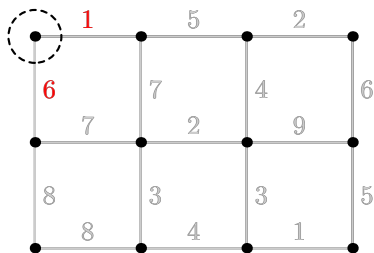
# Algorithme de Kruskal

Construire un arbre couvrant à partir d'un graphe sans arêtes.  
Considérer les arêtes du graphe dans un ordre croissant de leur poids. Sélectionner une arête si et seulement si elle ne forme pas de cycle avec des arêtes sélectionnées plus tôt.



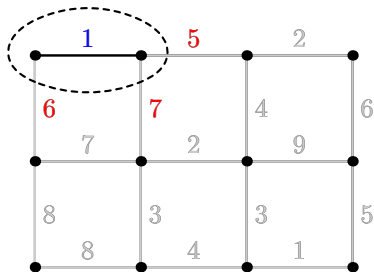
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



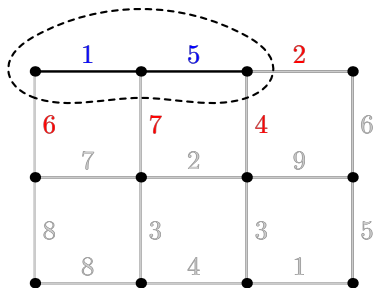
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



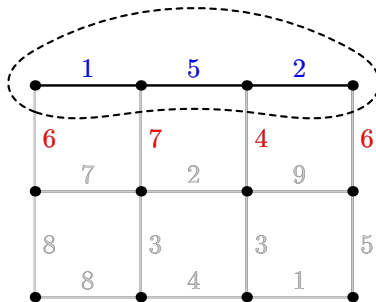
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



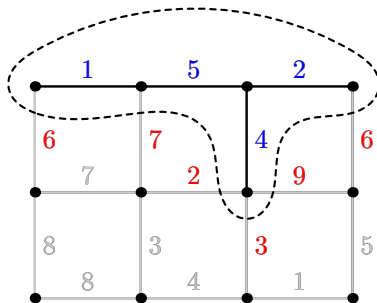
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



# Algorithme de Prim

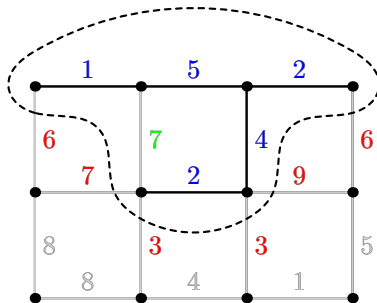
Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.





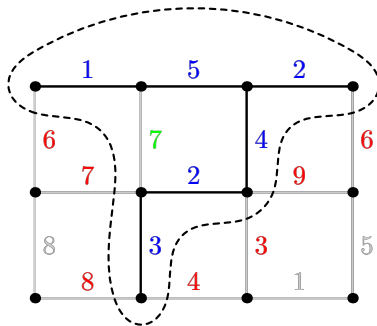
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



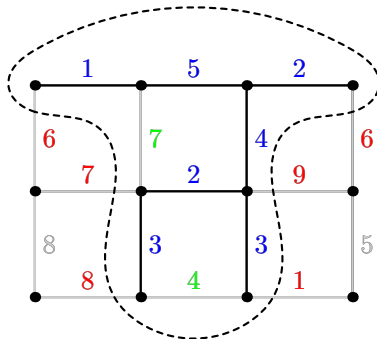
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



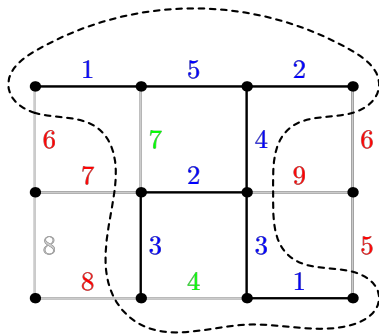
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



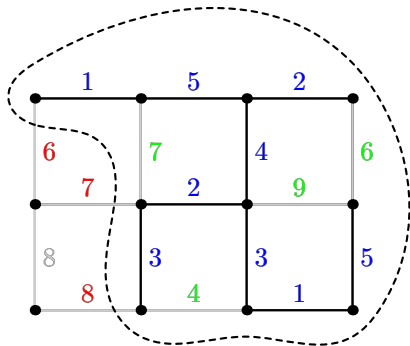
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



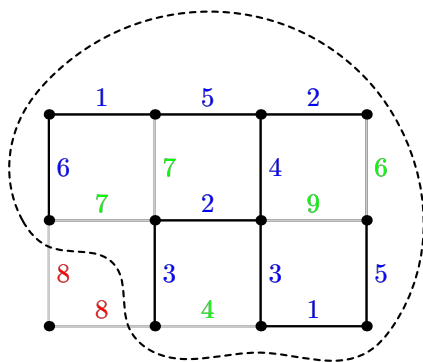
# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.



# Algorithme de Prim

Il y a un sommet de départ donné (la racine). À chaque étape, on ajoute l'arête la moins chère reliant un sommet (pour l'instant) isolé à un sommet déjà accessible depuis la racine.

