# AL5
# TD n° 8 : Arbres Couvrants Minimaux, algorithme de Prim

**Exercice 1 : arbres couvrants avec arêtes imposées**

Soit $G = (S, A)$ un graphe connexe.

**1.** Soit $(u, v) \in A$. Montrer qu'il existe un arbre couvrant qui contient $(u, v)$.

**2.** Soit $A' \subset A$. Donner une condition pour qu'il existe un arbre couvrant contenant $A'$.

On suppose maintenant que $G$ est muni d'une pondération $w : A \to \mathbf{R}$ de ses arêtes.

**3.** Soit $(u, v) \in A$. Existe-t-il toujours un arbre couvrant de poids minimal contenant $(u, v)$ ?

**4.** Même question si $(u, v)$ est une arête de poids minimum.

**Exercice 2 : algorithme de Prim**

Appliquer l'algorithme de Prim sur le graphe ci-dessous. On représentera la file de priorité à chaque itération.

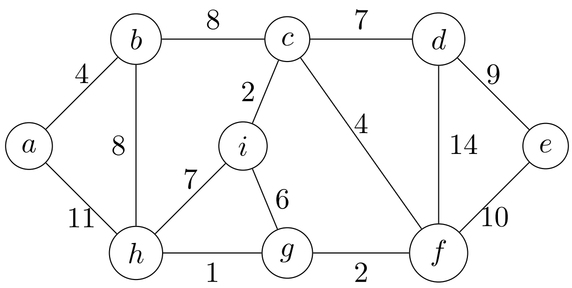


FIGURE 1 – graphe $G$ pour appliquer l'algorithme de Prim

**Exercice 3 : arbres couvrants de poids minimal**

Soit $G = (S, A, w)$ un graphe connexe pondéré. Soit $T$ un arbre couvrant de $G$. On note $A(T)$ l'ensemble des arêtes de $T$.

**1.** Montrer que si $a \in A \setminus A(T)$, il existe un unique cycle (que l'on notera par la suite $C_a$) dans le graphe $(S, A(T) \cup \{a\})$.

**2.** Montrer que les propositions suivantes sont équivalentes :

$(i)$ $T$ est un ACM de $G$.

$(ii)$ $\forall a \in (A \setminus A(T))\ \forall a' \in C_a\ \ w(a') \leqslant w(a)$.

$(iii)$ Pour **toute** partition $(X, S \setminus X)$ de l'ensemble des sommets, il existe une arête de $T$ qui traverse la partition et qui est de poids minimum parmi toutes les arêtes reliant un sommet de $X$ à un sommet de $S \setminus X$

*Indication : montrer* $(i) \implies (ii) \implies (iii) \implies (i)$. *Pour* $(iii) \implies (i)$*, on pourra montrer qu'il existe un déroulement de l'algorithme de Prim qui produit exactement* $T$.

**3.** En déduire que pout tout arbre couvrant $T = (S, A(T))$ non minimal, il existe une arête $a \in A \setminus A(T)$ et une arête $a' \in A(T)$ telle que $T^* = (S, A(T) \cup \{a\} \setminus \{a'\})$ est un arbre couvrant de poids strictement inférieur à celui de $T$.

**4.** Si $a$ est une arête de $G$, on note $G \setminus a$ le graphe pondéré donné par $(S, A \setminus \{a\}, w)$. En utilisant la question 2, montrer que si $a$ est une arête de poids maximum dans $G$ telle que le graphe $G \setminus a$ est encore connexe, alors tout ACM de $G \setminus a$ est aussi un ACM de $G$.

**5.** Déduire de la question précédente un algorithme qui part d'un graphe $G$ et construit un ACM en retirant une à une des arêtes du graphe. Sa complexité est-elle aussi bonne que celle de Prim ?

**Exercice 4 : ACM vs PCC : le cas "max"**

Ici on souhaite modifier les algorithmes de Dijkstra et de Prim pour essayer de calculer un arbre couvrant maximal (pour Prim) et les plus longs chemins depuis un sommet vers tous les autres (pour Dijkstra). Les modifications qu'on fait sont très simples :

- on utilise un tas prioritaire "max" (au lieu de "min") ;
- on initialise le tableau de distance avec $-\infty$ (au lieu de $\infty$) ;
- quand on examine $d(b) \overset{(?)}{\leqslant} d(a) + w(a, b)$ pour Dijkstra, on met à jour $d(b)$ si la nouvelle distance passant par $a$ est plus grande que la distance sauvegardée ;
- de même pour Prim, quand on examine $d(b) \overset{(?)}{\leqslant} w(a, b)$, on met à jour $d(b)$ si $w(a, b) > d(b)$.

**1.** L'algorithme de Prim ainsi modifié calcule-t-il un arbre couvrant maximal ?

**2.** L'algorithme de Dijkstra ainsi modifié calcule-t-il les longueurs des plus longs chemins depuis un sommet $s$ ?