AL5
TD n° 9 : algorithme de Kruskal

Exercice 1 : algorithme de Kruskal

Trouver un arbre couvrant de poids minimal sur les graphes suivants en appliquant l'algorithme de Kruskal.

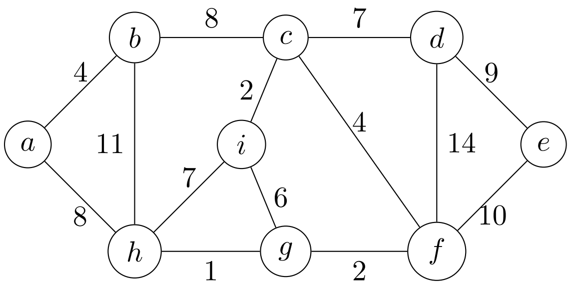

Figure 1 – Graphe G

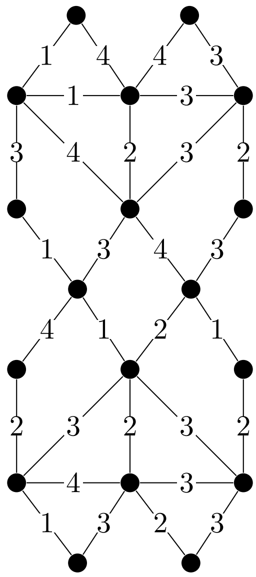

Figure 2 – Graphe H

Exercice 2 : ordre de parcours

Selon l'ordre dans lequel les arêtes de même poids sont examinées, l'algorithme de Kruskal ne retourne pas le même ACM. Pour un ACM donné, existe-t-il nécessairement un ordre tel que cet arbre soit la sortie de l'algorithme de Kruskal ?

Exercice 3 : unicité d'un ACM (encore)

1. Donner un exemple de graphe admettant plusieurs *arbres couvrants minimaux différents.*

2. Soit $T \subseteq A$ un arbre couvrant minimal de G, et $\{x, y\}$ une arête appartenant à T. On définit le graphe G' comme le graphe G dans lequel les sommets x et y ont été fusionnés en un nouveau sommet z.

Formellement, G' est le graphe (S', A', w') avec $S' = (S \backslash \{x, y\}) \cup \{z\}$ et les arêtes A' sont celles de $A \backslash \{(x, y)\}$ où les sommets x et y sont remplacés par z. La fonction w' renvoie le même poids que w. S'il existe $(x, u) \in A$ et $(y, u) \in A$, on ne garde dans G' qu'une seule arête (z, u), de poids $\min(w(x, u), w(y, u))$.

a. Choisir deux sommets du graphe exemple de la question précédente et dessiner le graphe G' correspondant.

b. Montrer que l'ensemble d'arêtes T' contenant les arêtes de $T \backslash \{(x, y)\}$ où les sommets x et y sont remplacés par z, est un arbre couvrant minimal du graphe G'.

3. Montrer que si la fonction poids w de $G = (S, A, w)$ est telle que le poids de chaque arête de A est unique, alors il existe un unique arbre couvrant minimal pour G.
Indice : on pourra utiliser le résultat précédent, faire une récurrence sur $|S|$ et montrer que l'arête de poids minimal est toujours présente dans un ACM de G...

Exercice 4 : structures Union-Find

Appliquer les opérations ci-dessous sur des structures Union-Find sur un ensemble d'entiers : $\{\mathsf{MakeSet}(i)\}_{i=1...10}$, $\mathsf{Union}(1, 2)$, $\mathsf{Union}(3, 4)$, $\mathsf{Union}(3, 2)$, $\mathsf{Union}(6, 5)$, $\mathsf{Union}(7, 8)$, $\mathsf{Union}(1, 7)$, $\mathsf{Union}(9, 10)$, $\mathsf{Union}(10, 5)$, $\mathsf{Find}(3)$, $\mathsf{Union}(6, 2)$, $\mathsf{Find}(10)$.

Exercice 5 : voyageur de commerce

On considère un graphe non-orienté valué $G = (S, A, w)$ avec $w : A \to \mathbf{R}_+$. On s'intéresse ici au problème du voyageur de commerce : on cherche une *tournée* (c-à-d. un circuit passant exactement une fois par chaque état de S) telle que la somme des arêtes de cette tournée soit minimale.
On s'intéresse au cas où A est complet (c-à-d. pour tout $u, v \in S$, on a $(u, v) \in A$) et G vérifie l'inégalité triangulaire : pour tous sommets u, v et x, on a : $w(u, v) \leqslant w(u, x) + w(x, v)$. Le problème du voyageur de commerce[1] (même dans le cas qui nous intéresse) est un problème considéré comme difficile (NP-complet).
On va utiliser la recherche d'un ACM pour G pour obtenir une approximation de la solution.

1. Soit C_{opt} un ensemble d'arêtes correspondant à une tournée minimale. Soit $T \subseteq A$ un ACM pour G. Montrer que l'on a : $w(T) \leqslant w(C_{opt})$.

2. On considère l'arbre (S, T) et un sommet $q \in S$. Soit P le parcours *complet* de l'arbre (S, T) : on part de q et on visite tous ses sous-arbres *etc.* et à la fin on revient à q. En déduire que le poids du parcours P est $2 \cdot w(T)$.
Proposer un parcours P depuis q_0 et vérifier la propriété prouvée sur l'exemple ci-dessous.

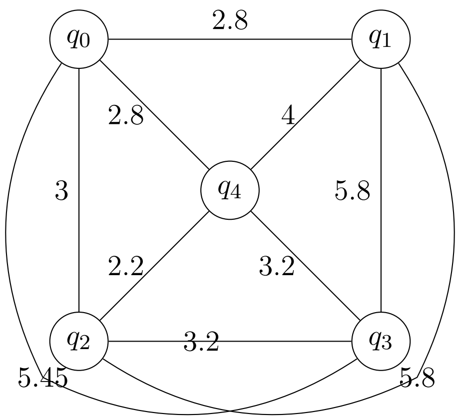

FIGURE 3 – exécution de l'algorithme de Prim (étapes 2 et 4).

3. On note C_A la liste des sommets visités lors d'un parcours préfixe de (S, T) à partir de $q \in S$. Montrer que C_A correspond à un circuit de G. Que peut-on déduire sur $w(C_A)$ comparé à $w(C_{opt})$?

1. sous sa forme de problème de décision : existe-t-il une tournée ayant un poids inférieur à une certaine constante donnée ?

Exercice 6 : arbre couvrant de poids min,$\times$

On se propose maintenant d'étudier le problème de trouver l'arbre couvrant de poids minimum quand la fonction de valuation d'un graphe $G = (S, A, \omega)$ est définie par :

$$\omega(G) = \prod_{(x,y) \in A} \omega(x, y)$$

On suppose que tous les poids sont entiers, positifs ou nuls excepté peut-être un poids qui peut être négatif.

1. Ce problème est-il relié à celui de l'ACM ?

2. Donner un algorithme pour trouver un tel arbre et calculer sa complexité.