# Fiche pédagogique

## Activité : Soyons connectés à moindre coût.

**Objectifs pédagogiques :** Se familiariser avec les notions de graphes, de chemins dans les graphes et d'algorithmes. Expérimenter et concevoir des algorithmes. Pour les plus grands, parler d'algorithmique distribuée (avec uniquement connaissance locale).

**Notions abordées :** Trier des nombres, additions, graphes, chemins, cycles, connexité, algorithme, algorithme distribué.

**Matériel nécessaire :** Papiers et crayons (on pourra dessiner les graphes) ou (version grandeur nature) des cerceaux (idéalement un par élève d'une demi-classe), des lattes ou des cordes (pour les arêtes) et des étiquettes (une par arête pour indiquer sa longueur).

**Niveau :** À partir du cycle 3.

**Durée :** Une demi-heure (peut être un peu plus avec des petits).

**Déroulement :**

*L'histoire* : On désire connecter les maisons d'un lotissement à la fibre (ou l'électricité, ou l'eau... ça n'a pas d'importance). Une des maisons est déjà connectée et le but est de décider où placer les fibres (c'est-à-dire de choisir quelles rues doivent être empruntées) de façon à ce que chaque maison soit reliée à la maison initiale. Cependant, chaque rue a une certaine longueur qui implique un certain coût d'installation de la fibre dans cette rue. L'objectif est de trouver une solution de coût total (la somme des longueurs des rues empruntées) minimum.

Ajouter une photo de l'activité

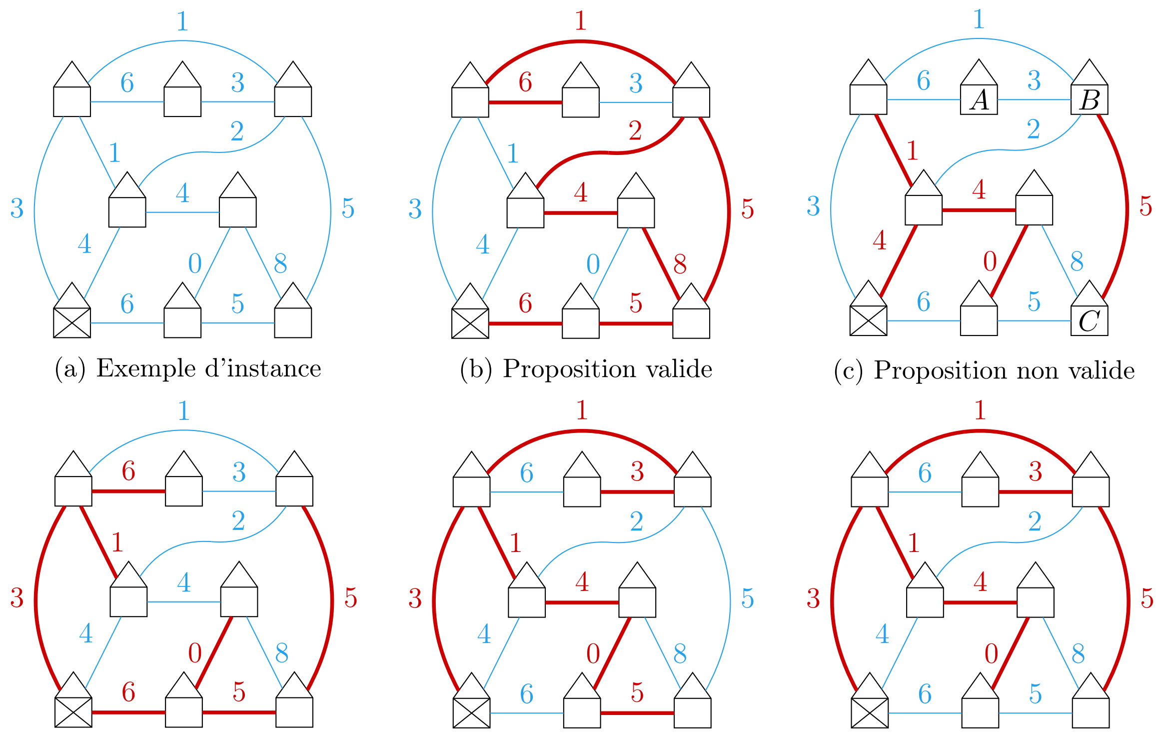
*Précisions par l'exemple* : Commencez par dessiner un graphe figurant le lotissement [1]. Le graphe doit être connexe, c'est-à-dire, qu'il doit exister un chemin (suivant les arêtes et qui passe possiblement par d'autres maisons) entre chaque paire de maisons. On décide également d'un poids (un nombre) pour chaque arête (sa longueur ou son coût). Un exemple est représenté sur la Figure **??**$a$) (ici, on a figuré les sommets par de petites maisons).

Une fois le but et les règles expliqués, présentez votre exemple à chaque élève (le même exemple pour tous) et laissez-les expérimenter, c'est-à-dire, essayer de proposer une solution (un ensemble de rues qui connectent toutes les maisons et de coût total le plus petit possible).

Dans un second temps, il est intéressant qu'ils comparent leurs solutions. Plusieurs remarques intéressantes devraient émerger :

— Il est intuitivement intéressant de choisir en priorité les arêtes les plus courtes (de moindre coût) ;

1. Rappelons qu'un graphe est composé de sommets, figurés par des cercles et représentant ici les maisons, et des arêtes figurées par des lignes reliant certaines paires de sommets et représentant ici les rues du lotissement.



(a) Exemple d'instance (b) Proposition valide (c) Proposition non valide

(d) Proposition valide (e) Solution optimale (f) Autre solution optimale

FIGURE 1 – La figure a représente une instance du problème (un exemple de lotissement) avec 8 maisons (les sommets) et 13 arêtes (représentées en bleu). La longueur (le coût) de chaque rue (arête) correspond au nombre bleu proche de l'arête (dans cet exemple, les coûts sont des entiers mais ils pourraient être n'importe quel nombre positif ou nul). La maison initiale (où arrive la fibre) est celle avec une croix. Les figures b à f sont des propositions de solutions : les arêtes choisies pour installer la fibre sont représentées en rouge. La proposition c n'est pas valide car les maisons $A, B$ et $C$ ne sont pas reliées à la maison initiale (avec une croix) par les fibres (en rouge). Les propositions b (de coût $1+2+4+5+5+6+6+8 = 37$) et d (de coût $0+1+3+5+5+6+6 = 26$) sont valides (toutes les maisons sont connectées) mais pas optimales (en particulier, la proposition b contient un cycle non nul). Les propositions e et f sont valides et optimales : de coût minimum $0+1+1+3+3+4+5 = 17$.

— Toute solution optimale (de coût minimum) ne contient aucun cycle dont les arêtes ont toutes des poids strictement positifs. Par exemple, la proposition de la Figure 1b contient un cycle avec des arêtes de poids $2, 4, 8$ et $5$. Dans ce cas, supprimer l'arête de poids maximum du cycle (ici celle de poids 8) donne une solution valide de poids inférieur 29 ;

— Toute solution optimale (de coût minimum) contient exactement $n-1$ arêtes (où $n$ est le nombre de sommets/maisons).

(cette dernière propriété est peut-être plus difficile à remarquer)

*Un autre exemple d'application* : On peut réitérer le problème précédent avec un exemple de graphe où toutes les arêtes ont le même poids. Dans ce cas, tout arbre couvrant (ensemble d'arêtes connectant tous les sommets et sans cycle) est une solution optimale. Notez, qu'en général, il peut exister plusieurs solutions optimales.

*Un algorithme optimal* (Kruskal) : Commencez par ordonner les arêtes par poids croissants (de la plus petite à la plus grande). Les arêtes de poids identiques sont ordonnées arbitrairement. Sur l'exemple : $(0, 1, 1, 2, 3, 3, 4, 4, 5, 5, 6, 6, 8)$. En parcourant les arêtes dans cet ordre ("du plus petit au plus grand"), à chaque nouvelle arête rencontrée, on l'ajoute à la solution courante si elle ne crée pas de cycle, et on l'élimine de la solution sinon. On s'arrête lorsque tous les sommets sont connectés (ou, de façon équivalente, lorsque $n - 1$ arrêtes ont été sélectionnées)

Un exemple de déroulement de cet algorithme (Algorithme de Kruskal) est donné dans la Figure **??**.

L'algorithme présenté calcule une solution valide (qui connecte toutes les maisons les unes aux autres). La preuve que la solution calculée est optimale (de coût minimum) est au-delà de cette activité mais on peut remarquer que la solution calculée est un arbre couvrant (i.e., connecte toutes les maisons et sans cycle).

*Version grandeur nature* : La version grandeur nature peut être déclinée de la même façon que la version décrite jusqu'ici avec, à la place d'un graphe dessiné sur du papier ou un tableau, un graphe construit avec des cerceaux et des lattes. Au niveau fin collège-lycée, il peut être intéressant d'ajouter des bases d'algorithmique distribuée : chaque élève se trouve dans un cerceau, ne connait que les poids des arêtes qui lui sont adjacentes et ne peut communiquer qu'avec les élèves dans les cerceaux voisins (reliés par une arête). Comment, avec ces contraintes, implémenter l'algorithme précédent ? En particulier, comment détecter les cycles ? comment savoir quand toutes les maisons ont été connectées ?

**Aller plus loin :** Au niveau fin Lycée, on peut prouver (par récurrence sur le nombre de sommets) qu'un arbre (graphe connexe acyclique) de $n$ sommets a exactement $n - 1$ arêtes (Prouver qu'un arbre a au moins un sommet $v$ avec un unique voisin, montrer que si l'on supprime ce sommet, ce qui reste est un arbre et appliquer l'induction).

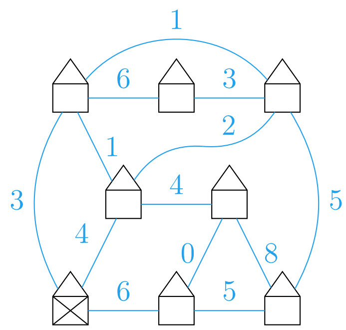
Prouver que l'algorithme présenté (de Kruskal) est optimal (calcule une solution de poids minimum) n'est pas trivial, mais peut être expliqué fin Lycée.

*Conclusion* : Un algorithme glouton "simple" peut parfois permettre de résoudre un problème *a priori* compliqué (ici, le problème d'arbre couvrant minimum dans un graphe dont les arêtes sont pondérées).

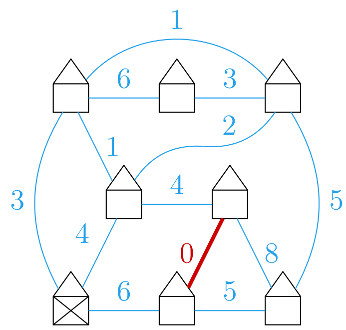
**Référence :**

— `https://fr.wikipedia.org/wiki/Arbre_couvrant_de_poids_minimal`

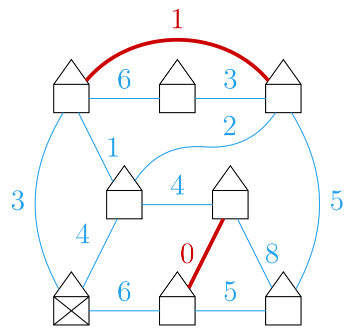
**Contact :** nicolas point nisse arobase inria point fr



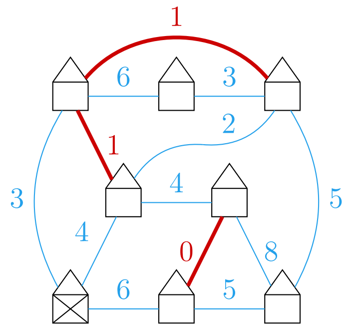
(a) Exemple d'instance



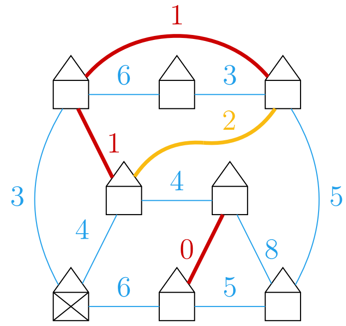
(b) Ajout d'une arête de poids 0



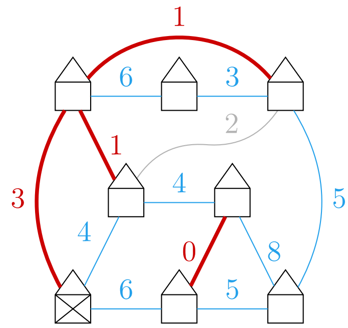
(c) Ajout d'une arête de poids 1



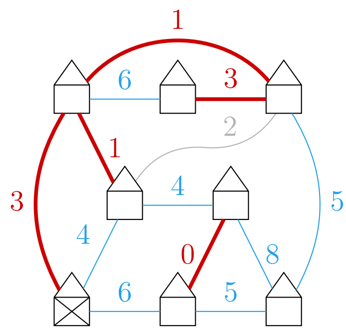
(d) Ajout d'une arête de poids 1



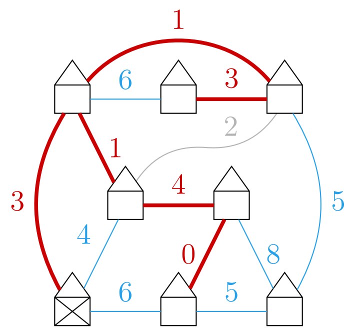
(e) Une arête de poids 2 est rejetée



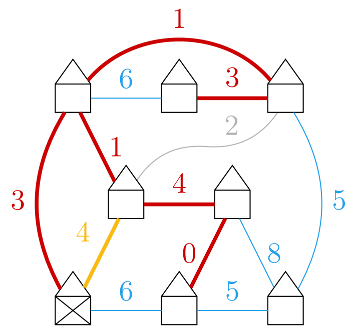
(f) Ajout d'une arête de poids 3



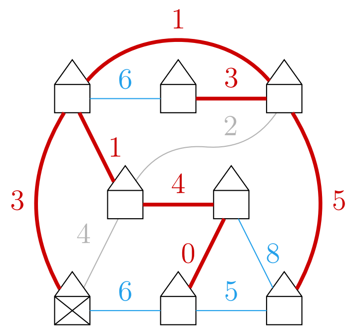
(g) Ajout d'une arête de poids 3



(h) Ajout d'une arête de poids 4



(i) Une arête de poids 4 est rejetée



(j) Ajout d'une arête de poids 5
Toutes les maisons sont connectées



FIGURE 2 – Exemple d'application de l'algorithme de Kruskal (décrit plus haut).