Algorithmique et Programmation
TD n°8 : Flots

Ecole normale supérieure - Département d'informatique
algoL3@di.ens.fr

2015-2016

Exercice 1 — Minimum Edge Separator

1. Donner un algorithme qui calcule l'ensemble d'arêtes minimum qui sépare deux sommets s et t d'un graphe non-orienté.
2. Proposer un algorithme qui calcule un minimum edge separator, c'est-à-dire un ensemble d'arêtes de taille minimum dont le retrait déconnecte le graphe non-orienté donné en argument (supposé connexe).

Exercice 2 — Mise à jour de flots

Soit $G = (V, E)$ un réseau de transport de capacités entières et soit f un flot maximum de G.

1. Supposons que la capacité d'une arête $(u, v) \in E$ est augmentée de 1. Donner un algorithme de complexité $\mathcal{O}(E)$ qui met à jour le flot maximum f.
2. Supposons que la capacité d'une arête $(u, v) \in E$ est diminuée de 1. Donner un algorithme de complexité $\mathcal{O}(E)$ qui met à jour le flot maximum f.

Exercice 3 — Plan d'évacuation

On considère une grille de taille $n \times n$, c'est-à-dire un graphe de n^2 sommets, où chaque sommet est reliée à ses voisins horizontaux et verticaux. Chaque sommet a donc 4 voisins, sauf ceux qui sont sur le bord de la grille (on peut imaginer qu'il s'agit d'une modélisation grossière d'une salle de spectacle).

Chaque sommet peut être ou bien *libre* ou bien *occupé*. Dans ce contexte, un plan d'évacuation est un ensemble de chemins *sans sommets commun*, passant uniquement par des sommets libres. Chaque chemin doit partir d'un sommet occupé, et atteindre un sommet du bord de la grille.

Donner un algorithme polynomial qui détermine si un plan d'évacuaton existe et, si c'est le cas, qui le calcule.

Exercice 4 — Extension d'un problème de flots

On s'intéresse aux problèmes de flots avec offre et demande. On peut imaginer que le graphe décrit une carte, que les sommets sont des villes, et que le flot modélise le transport d'une marchandise donnée (par exemple, des vis). Les ville produise et consomment des vis en quantité déterminée. Chaque sommet v du graphe est donc étiqueté avec un réel $d(v)$ qui précise la demande en vis. Si $d(v)$ est supérieur (resp. inférieur) à zéro, alors la ville v consomme (resp. produit) $d(v)$ unités de vis par unité de temps. Une circulation est un flot qui respecte les contraintes de capacité des arêtes, et où la contrainte de conservation du flot est transformée en :

$$\forall v \in V, \quad \sum_{(u,v)\in E} f(u, v) - \sum_{(v,u)\in E} f(v, u) = d(v)$$

1. Énoncer une condition nécessaire simple sur les demandes pour qu'une circulation existe.
2. Transformer le problème en un problème de flot ordinaire (en introduisant une source et un puit).
3. Donner un algorithme polynomial qui détermine si une circulation existe.
4. Démontrer que si les capacités et les demandes sont entières, et qu'il existe une circulation, alors il en existe une entière.