Université de Strasbourg **UFR de Mathématiques et Informatique**
L2 Informatique et CMI S4 **2017-2018, semestre de printemps**

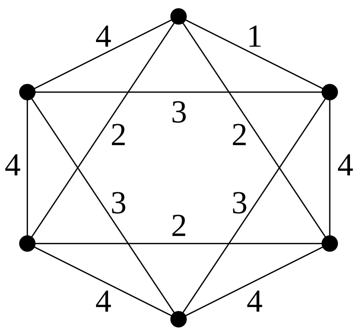
Graphes

Contrôle Continu n°3, 11 mai 2018

Durée : 1 heure 30 minutes
Responsable : Prof. Christian RONSE
Tous documents en papier autorisés mais non partagés
Calculettes inutiles
Téléphones et appareils électroniques éteints et rangés dans un sac fermé

Justifiez soigneusement vos réponses !

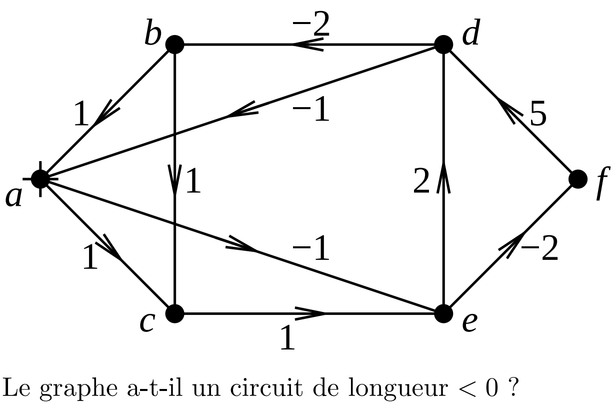
(1) Arbres couvrants de poids minimum.



Dans le graphe non orienté pondéré ci-contre (où le poids de chaque arête est indiqué par le nombre à côté d'elle), construire un arbre couvrant de poids minimum par l'algorithme de Kruskal. *Il faut donner toutes les étapes de la construction.*

Ensuite expliquer quels sont les différents choix possibles pour un arbre couvrant de poids minimum.

(2) Arborescence et distances depuis une racine par Bellman-Ford.
Dans le graphe orienté pondéré ci-contre, on prend la racine a (marquée d'une croix). Construire l'arborescence et les distances de la racine à tous les sommets au moyen de l'algorithme de Bellman-Ford, où à chaque étape l'ordre d'examen (et de relaxation) des arcs est alphabétique sur les origines puis sur les buts :



$$ac\ ;\ ae\ ;\ ba\ ;\ bc\ ;\ ce\ ;$$
$$da\ ;\ db\ ;\ ed\ ;\ ef\ ;\ fd\ .$$

(Pour simplifier, un arc d'origine x et de but y est ici écrit xy). *Il faut donner toutes les étapes de la construction.*

Le graphe a-t-il un circuit de longueur < 0 ?

(3) Questions sur l'algorithme de Floyd-Warshall.
Pour un graphe orienté à n sommets $s_1, \ldots, s_n$, l'algorithme de Floyd-Warshall calcule la distance

(pondérée) $d(i,j)$ de s_i à s_j $(i,j=1,\dots,n)$ comme suit. Pour $k=0$ on pose

$$D^{(0)}(i,i)=0 \quad \text{et pour } i\neq j,\ D^{(0)}(i,j)=\begin{cases} p(i,j) & \text{s'il y a un arc de poids } p(i,j) \text{ de } s_i \text{ vers } s_j, \\ \infty & \text{s'il n'y a pas d'arc de } s_i \text{ vers } s_j\ . \end{cases}$$

Ensuite pour $k=1,\dots,n$, on calcule $D^{(k)}(i,j)$ par récurrence sur k selon la formule

$$D^{(k)}(i,j)=\min\Big(D^{(k-1)}(i,j), D^{(k-1)}(i,k)+D^{(k-1)}(k,j)\Big)\ .$$

À la fin, on obtient $d(i,j)=D^{(n)}(i,j)$.

En fait $D^{(k)}(i,j)$ donne la longueur minimum d'un chemin de s_i à s_j dont tous les sommets intermédiaires appartiennent à $\{s_t \mid 1\leq t\leq k\}$ (donc sans aucun sommet intermédiaire pour $k=0$), s'il en existe, et ∞ s'il n'en existe pas.

Questions : On veut simplifier le calcul des $D^{(k)}(i,j)$.

(i) Pour quels triples (i,j,k) peut-on ramener $D^{(k)}(i,j)$ à une valeur “déjà connue”, c.-à-d. une constante ou un $D^{(x)}(u,v)$ pour $x<k$?

(ii) Même question dans le cas particulier où le graphe est sans circuit et trié topologiquement, c.-à-d. pour un arc d'origine s_i et de but s_j on a nécessairement $i<j$.