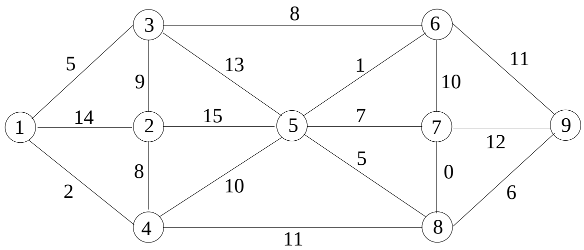
Universität Erlangen-Nürnberg
Naturwissenschaftliche Fakultät I
Sommersemester 2004
Prof. Dr. K. Klamroth
Barbara Pfeiffer

30. 04. 2004

Netzwerkoptimierung
Übungsblatt 2

Problem 1:
Gegeben sei der folgende Graph $G = (N, A)$:



Bestimmen Sie einen minimalen spannenden Baum von G

(a) mit Hilfe von Kruskal's Algorithmus

(b) mit Hilfe von Prim's Algorithmus

(c) mit Hilfe von Sollin's Algorithmus.

Problem 2:
Beweisen Sie die folgenden Optimalitätsbedingungen für *maximale spannende Bäume*, d.h. Bäume, deren Kosten maximal sind:

(a) $T^* = (N, A(T^*))$ ist ein maximaler spannender Baum von $G = (N, A)$
$\Leftrightarrow$ Für alle Kanten $[i, j] \in A(T^*)$ gilt:
$c_{ij} \geq c_{kl}$ für alle Kanten $[k, l] \in A$, die in dem Schnitt von G enthalten sind, der durch Entfernen der Kante $[i, j]$ aus T^* definiert ist.

(b) $T^* = (N, A(T^*))$ ist ein maximaler spannender Baum von $G = (N, A)$
$\Leftrightarrow$ Für alle Kanten $[k, l] \in A \setminus A(T^*)$ gilt:
$c_{ij} \geq c_{kl}$ für alle Kanten $[i, j] \in A(T^*)$, die in dem Weg von k nach l in T^* enthalten sind.

Problem 3:
Sei $T = (N, A(T))$ ein spannender Baum von $G = (N, A)$. Für $i, j \in N$ sei mit $\beta[i, j]$ diejenige Kante in T bezeichnet, die die minimalen Kosten unter allen Kanten hat, die auf dem Weg von i nach j in T liegen. Geben Sie einen (effizienten!) Algorithmus an, mit dem man $\beta[i, j]$ für alle Paare $[i, j] \in N \times N$ bestimmen kann.

Problem 4:
Sei $T^* = (N, A(T^*))$ ein minimaler spannender Baum von $G = (N, A)$. Für jede Kante $[i, j] \in A$ wird als *Kostenintervall* diejenige Menge aller Gewichte $c_{ij} \geq 0$ bezeichnet, für die T^* ein minimaler spannender Baum bleibt.

(a) Geben Sie einen (effizienten!) Algorithmus an, der das Kostenintervall für eine gegebene Kante $[i, j] \in A$ bestimmt.

(b) Geben Sie einen Algorithmus an, der das Kostenintervall für alle Kanten $[i, j]$ bestimmt, und der effizienter ist als die Anwendung der Methode aus (a) auf alle Kanten $[i, j] \in A$.