Technische Universität München, Zentrum Mathematik
Lehrstuhl für Angewandte Geometrie und Diskrete Mathematik

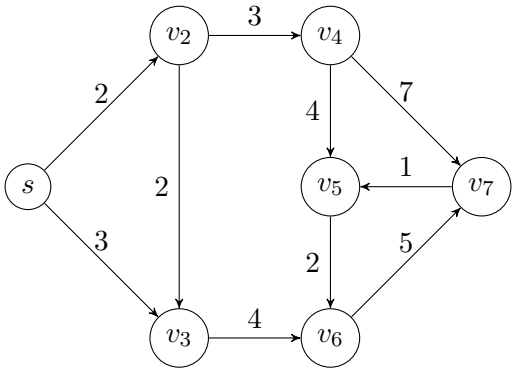
Algorithmische Diskrete Mathematik (MA 2501)

Prof. Dr. P. Gritzmann | Dr. M. Ritter | Dipl.-Math. M. Silbernagl

Aufgabenblatt 7

Tutoraufgabe 7.1 (Kürzeste Wege) [4]

Berechnen Sie mit Hilfe des Floyd-Warshall-Algorithmus im untenstehenden Digraphen $G = (V, E)$ die Längen aller kürzesten v_i-v_j-Wege für alle Knoten $v_i, v_j \in V$. Die Zahlen an den Kanten geben jeweils die Kantengewichte an. Stellen Sie Ihre Rechnung so dar, dass die Einzelschritte nachvollziehbar sind.



Aufgabe 7.2 (Floyd-Warshall – Modellierung einer Währungsbörse) [4+1]

a) Passen Sie den Floyd-Warshall-Algorithmus aus der Vorlesung möglichst einfach an, sodass man aus seiner Ausgabe in Laufzeit $\mathcal{O}(n)$ entweder

- einen negativen Kreis bestimmen kann, falls der Algorithmus mit der Meldung „Kreis negativer Länge“ endet, oder
- für ein beliebiges Knotenpaar v_i, v_j einen kürzesten v_i-v_j-Weg konstruieren kann.

Für jedes Knotenpaar den momentan besten Weg explizit zu speichern gilt *nicht* als möglichst einfach.

b) Sie sind ein Händler an einer Währungsbörse und haben einen Insidertipp erhalten: Durch einen Systemfehler ist es gerade möglich, risikofrei Gewinn zu erwirtschaften.

Die gehandelten Währungen sind als Menge W und die Wechselkurse als Funktion $k(w, w')$ gegeben. Für eine Einheit der Währung w erhalten Sie also $k(w, w')$ Einheiten der Währung w'. Alle Währungen sind ineinander tauschbar, und es sind beliebig viele Währungsumtausche zulässig. Steuern und Abgaben sind bereits einberechnet.

Entwickeln Sie eine Algorithmus, der Folgendes leistet: Nach Eingabe aller Währungen W und aller Wechselkurse ermittelt der Algorithmus eine Währung $w_0 \in W$ und eine Tauschvorschrift, sodass Sie nach Anwendung der Tauschvorschrift mehr Geld in Währung w_0 besitzen als vorher. Schulden in anderen Währungen sind dabei natürlich nicht erlaubt.

Bitte wenden!

Aufgabe 7.3 (Kürzeste Kreise) [3]
Sei $G = (V, E, \phi)$ ein gewichteter Digraph, der keine Kreise negativer Länge enthalte. Modifizieren Sie den Floyd-Warshall-Algorithmus so, dass für jeden Knoten die Länge eines kürzesten ihn enthaltenden Kreises bestimmt wird (bzw. festgestellt wird, dass der Knoten in keinem Kreis enthalten ist).

Aufgabe 7.4 (Steinerbäume mit 2 oder 3 Terminalknoten) [2+3]
Sei $G = (V, E; \phi)$ ein ungerichteter Graph mit nichtnegativen Kantengewichten ϕ und $T \subset V$. Ein Baum $ST \subset G$ heißt *Steinerbaum* in G, wenn er alle Knoten aus T enthält. Man bezeichnet T als *Terminalknoten* und $V(ST) \setminus T$ als *Steinerknoten*. Das Steinerbaum-Problem besteht darin, zu gegebenem G und T einen Steinerbaum ST mit minimalem Kantengewicht $\phi(E(ST))$ zu finden.

Entwerfen Sie möglichst effiziente Algorithmen, die das Steiner-Baum-Problem für $|T| = 2$ bzw. $|T| = 3$ lösen. Welche Komplexität haben Ihre Algorithmen in Abhängigkeit von $n = |V|$?

Aufgabe 7.5 (Ford-Fulkerson-Algorithmus) [6]
Zur Bestimmung eines maximalen Flusses zwischen zwei Knoten s und t in einem Netzwerk N kann man wiederholt einen beliebigen Augmentationsweg auswählen und entlang dieses Weges den Fluss so weit erhöhen, bis für eine Kante des Weges ihre Kapazitätsgrenze erreicht ist.

Konstruieren Sie für beliebiges $C \in \mathbb{N}$ ein Netzwerk N mit maximal 8 Kanten und Kapazitäten aus $\{1, 2, \ldots, C\}$, in welchem ein solcher Algorithmus $\Omega(C)$ Flusserhöhungen benötigen kann, bis er einen maximalen s-t-Fluss gefunden hat.

Die Abgabedaten von Blatt 7 für Ihre Gruppe finden Sie auf der Homepage zur Vorlesung unter `http://www-m9.ma.tum.de/WS2012/ADM`.