Institut für Informatik I
Prof. Dr. Heiko Röglin
Tobias Brunsch

Algorithmen und Berechnungskomplexität I
Wintersemester 2010/11
Abgabe: 12.01.2011 in der Vorlesung

Übungsblatt 9

Aufgabe 9.1

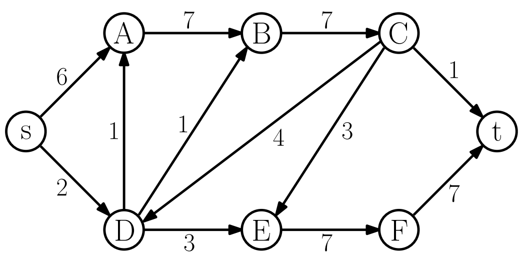
6 Punkte

Sei $G = (V, E)$ ein gerichteter Graph mit einer Kantengewichtung w. Wir führen den Algorithmus von Dijkstra für einen Knoten $s \in V$ aus. Ohne Einschränkung sei jeder Knoten des Graphen von s aus erreichbar. Mit v_i bezeichnen wir den Knoten von G, der als i-ter Knoten zur Menge S hinzugefügt wird. Zeigen Sie induktiv, dass $\delta(s, v_i) \leq \delta(s, v_{i+1})$ für alle $i = 1, \ldots, n-1$ gilt, d.h. zu keinem Zeitpunkt liegt ein Knoten außerhalb von S näher an s als irgendein Knoten in S. Nutzen Sie dabei die Invariante aus der Vorlesung.

Aufgabe 9.2

3+3 Punkte

(a) Bestimmen Sie in dem unten abgebildeten Netzwerk einen maximalen Fluss von Knoten s nach Knoten t. Begründen Sie, warum der von Ihnen angegebene Fluss maximal ist.



(b) Im Algorithmus von Ford und Fulkerson ist nicht vorgeschrieben, welcher augmentierende Pfad zu wählen ist, falls mehrere existieren. Zeigen Sie, dass die Laufzeit des Algorithmus von Ford und Fulkerson nicht in der Größe des Flussnetzwerkes beschränkt ist.

Hinweis: Geben Sie dafür ein Flussnetzwerk mit konstant vielen Knoten und Kanten an und zeigen Sie für dieses Netzwerk, dass die Anzahl der Verbesserungsschritte beliebig groß werden kann, wenn beliebige Kapazitäten erlaubt sind.

Aufgabe 9.3

3+3 Punkte

Führen Sie die folgenden Varianten des Flussproblems auf die Standardversion aus der Vorlesung zurück.

(a) Zusätzlich zu den Kantenkapazitäten gibt es Knotenkapazitäten. Für einen zulässigen Fluss f muss nun außerdem gelten:

$$\sum_{u\,:\,(u,v)\in E} f(u,v) \leq c(v) \text{ für alle } v \neq s \quad \text{und} \quad \sum_{v\,:\,(s,v)\in E} f(s,v) \leq c(s)\,,$$

wobei $c \colon V \longrightarrow \mathbb{N}_0$ die Kapazitäten der Knoten angibt.

(b) Statt einer Quelle s und einer Senke t gibt es jetzt mehrere Quellen $s_1, \ldots, s_k$ und mehrere Senken $t_1, \ldots, t_l$.

Aufgabe 9.4

6 Punkte

Gegeben sei die Tabelle der Fußball-Bundesliga, die den Punktestand zu einem bestimmten Zeitpunkt wiedergibt, sowie eine Liste der noch ausstehenden Spiele. Wir betrachten die alte Zweipunkteregel, d.h. die siegreiche Mannschaft erhält zwei Punkte und der Verlierer keinen. Bei einem Unentschieden erhält jede Mannschaft einen Punkt. Das *Meisterschaftsproblem* besteht darin zu entscheiden, ob eine gegebene Mannschaft noch Meister werden kann. Dazu muss sie in der Gesamtwertung die meisten Punkte haben. Gibt es mehrere Mannschaften mit derselben Punktzahl, dann muss sie unter diesen das beste Torverhältnis haben. Wir gehen davon aus, dass die gegebene Mannschaft noch mindestens ein Spiel zu bestreiten hat.

Modellieren Sie das Meisterschaftsproblem als Flussproblem.

Hinweis: Sie können annehmen, dass die gegebene Mannschaft alle ausstehenden Spiele gewinnt und dabei hinreichend viele Tore schießt, sodass sie von allen Mannschaften das beste Torverhältnis hat. So kann man berechnen, wie viele Punkte sie noch erreichen kann und es ist klar, wie viele Punkte jede der anderen Mannschaften höchstens noch bekommen darf. Es müssen also nur die Spiele betrachtet werden, an denen die Mannschaft nicht beteiligt ist. Die Frage ist demnach, ob die restlichen Spiele alle so ausgehen können, dass keine Mannschaft mehr als diese Punktzahl erreicht.

Aufgabe 9.5

6 Zusatzpunkte

Arbitrage bezeichnet das Ausnutzen von Preisunterschieden für gleiche Waren auf verschiedenen Märkten. Ein einfaches Beispiel ist das Ausnutzen von Wechselkursen, also das Handeln mit Währungen. Angenommen, ein Händler startet mit 1\$ (Dollar). Damit kauft er zunächst 95.739¥ (Yen). Pro Yen erhält er 0.0063£ (britische Pfund). Schließlich kauft er zum Kurs 1.6583\$/£ wieder Dollar und hat jetzt $95.739 \cdot 0.0063 \cdot 1.6583 = 1.0002131$ Dollar (Wechselkurse vom 02.06.2009). Insgesamt ergab sich also ein Gewinn von 0.021%.

Gegeben seien n Währungen $c_1, \ldots, c_n$ und eine $(n \times n)$-Matrix R, die die aktuellen Wechselkurse enthält. Geben Sie einen Algorithmus an, der für eine Wechselkursmatrix R in Zeit $O(n^k)$ bestimmt, ob ein Arbitrage-Geschäft möglich ist. Dabei soll k eine geeignete Konstante sein.