Abteilung Algorithmik
Institut für Betriebssysteme und Rechnerverbund **SoSe 15**
TU Braunschweig

Prof. Dr. Sándor P. Fekete
Frank Quedenfeld

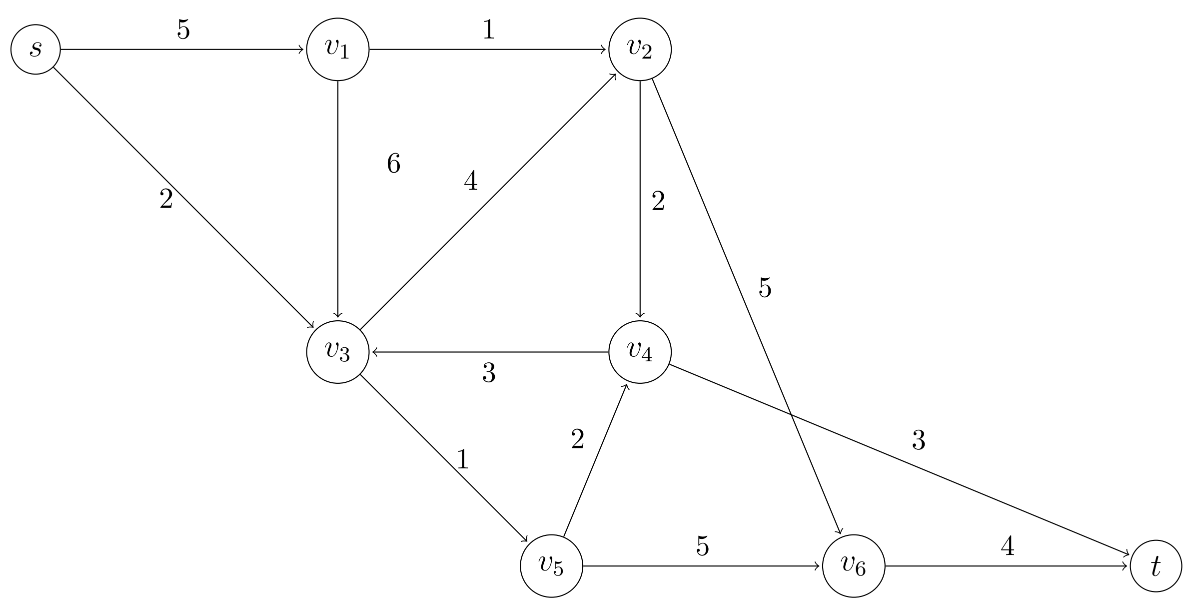
Netzwerkalgorithmen
Übung 5 vom 22.06.2015

Abgabe der Lösungen bis Montag,
den 06.07.15, 13:00 Uhr im Hausaufgabenrückgabeschrank.
Bitte die Blätter zusammenheften und vorne deutlich mit eigenem Namen,
Matrikel- und Gruppennummer, sowie Studiengang versehen!

Aufgabe 1 (Klausurvorbereitung):
Gib Deinen Namen (Format: Nachname, Vorname), Matrikelnummer und Studiengang (mit Zusatz Bachelor, Master, Diplom!) *leserlich* an.
Diese Angaben brauchen wir für die Weiterleitung der Klausurergebnisse, also gebt Euch Mühe `;-)`.

(2 Punkte)

Aufgabe 2 (Algorithmus von Ford und Fulkerson):



Bestimme mit Hilfe des Algorithmus von Ford und Fulkerson einen maximalen s-t-Fluss im Netzwerk (G, u, s, t). Gib jeweils den Residualgraphen an.
Gib außerdem einen minimalen Schnitt an.

(13+2 Punkte)

Aufgabe 3 (Fibonacci-Heaps):

Wir definieren für einen Knoten x in einem Fibonacci-Heap $degree[x]$ als die Anzahl seiner Kinder. Beweise:
Sei x ein Knoten in einem Fibonacci-Heap und nimm an, dass $degree[x] = k$ gilt. Seien $y_1, y_2, \ldots, y_k$ die Kinder von x, in der Reihenfolge, wie sie mit x verbunden wurden, vom frühesten zum spätesten. Dann gilt $degree[y_1] \geq 0$ und $degree[y_i] \geq i - 2$ für $i = 2, 3, \ldots, k$.
(Hinweis: Verliert ein Knoten ein Kind, verbleibt er im Fibonacci-Heap; verliert er ein zweites Kind, wird er abgeschnitten.)

(13 Punkte)

Aufgabe 4 (Kantendisjunkte Pfade):
Sei $G = (V, E)$ ein gerichteter Graph, $s, t \in V$ und $k \in \mathbb{N}$. Zwei Pfade P und Q heißen kantendisjunkt, wenn sie keine gemeinsame Kante haben.
Zeige: Es gibt genau dann k kantendisjunkte s-t-Pfade in G, wenn es nach dem Entfernen von (beliebigen) $k - 1$ Kanten aus G noch einen s-t-Pfad gibt.
(Tipp: Wende Max Flow = Min Cut auf ein geeignetes Netzwerk an. Verwende außerdem den Satz über die Dekomposition von Flüssen aus der Vorlesung)

(15 Punkte)

Aufgabe 5 (Perfektes Matching in bipartiten Graphen):
Ein perfektes Matching $M \subseteq E$ ist eine Menge von paarweise nicht-adjazenten Kanten, wobei zu jedem Knoten *genau eine* dieser Kanten inzident sein muss. Zeige, in einem bipartiten Graphen $G = (V, E)$ mit $V = A \cup B$ in dem jeder Knoten *genau* Grad k mit $k \geq 1$ hat, gibt es ein perfektes Matching. Benutze hierzu die Flussformulierung für bipartites Matching und argumentiere über die Größe eines minimalen s-t-Schnittes. Wende danach Max Flow = Min Cut an.

(15 Punkte)