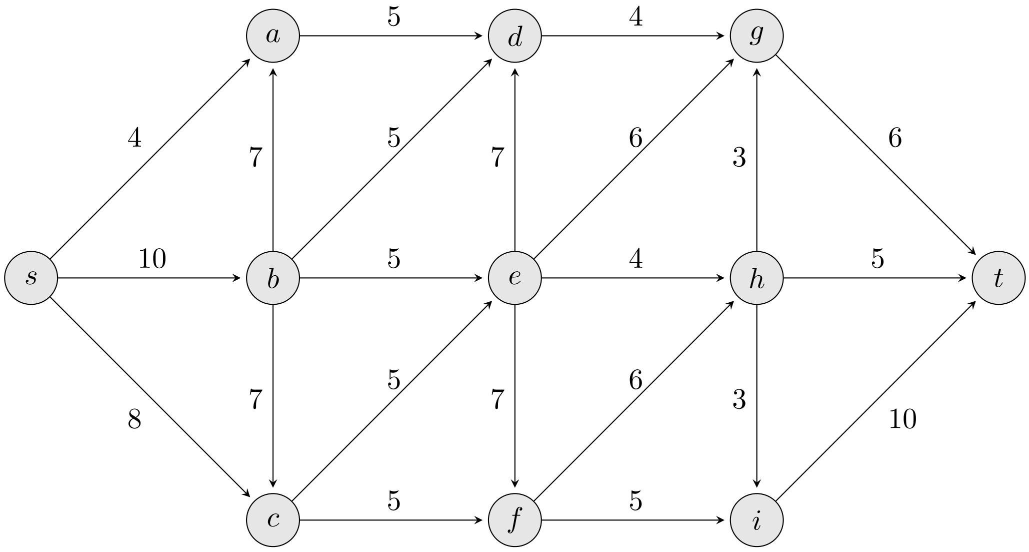
Aufgabenblatt 10.

Abgabedatum: **13.01.2016**.

Aufgabe 1. (Flussnetzwerke)



a) Berechnen Sie für das oben stehende Flussnetzwerk einen Maximalfluss f. Die angegebenen Zahlen geben die Kapazität der jeweiligen Kante an. Begründen Sie, dass Ihr Fluss wirklich maximal ist.

b) Geben sie einen Schnitt minimaler Kapazität an.

(5 Punkte)

Aufgabe 2. (Vitale Kanten)

Sei $N = (G, c, s, t)$ ein Fluß-Netzwerk.

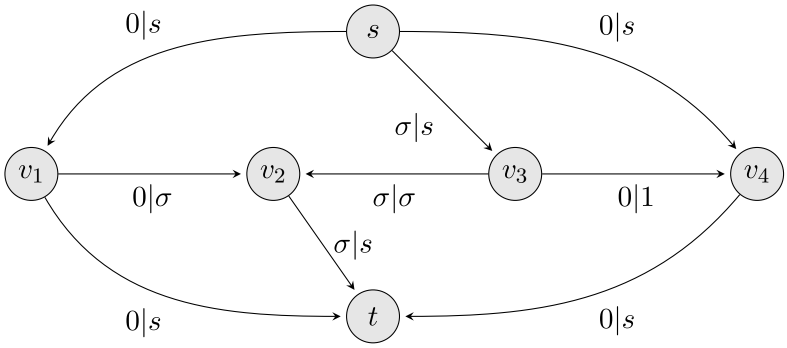
a) Man zeige, dass es mindestens eine Kante in N gibt, bei deren Wegnahme der maximale Fluss verkleinert wird.

b) Eine Kante e heißt *vital*, wenn durch die Entfernung der maximale Flusswert um den maximal möglichen Betrag verkleinert wird. Ist eine Kante maximaler Kapazität in einem minimalen Schnitt notwendig vital? Begründen Sie dies anhand eines Beweises oder eines Gegenbeispiels.

(5 Punkte)

Aufgabe 3. (Ford–Fulkerson)

Gegeben sei folgendes Netzwerk mit Fluss f:



Dabei sei $\sigma = \frac{\sqrt{5}-1}{2}$ und $s = \frac{1}{1-\sigma}$[1]. Der gegebene Fluss resultiert beispielsweise aus dem Nullfluss durch Augmentierung entlang des Wegs $\pi = s, v_3, v_2, t$. Zeigen Sie, dass es eine Folge von augmentierten Wegen $\pi_1, \pi_2, \pi_1, \pi_3, \pi_1, \pi_2, \pi_1, \pi_3, \ldots$ gibt, so dass der Ford-Fulkerson-Algorithmus nicht terminiert. Dazu beachte man, dass $\sigma^n = \sigma^{n+1} + \sigma^{n+2}$ gilt.

(5 Punkte)

Aufgabe 4. (Kapazitätsbeschränkte Knoten)

Es sei ein Fluss-Netzwerk $N = (G, c, s, t)$ gegeben. Wir wollen nun den Fall von kapazitätsbeschränkten Knoten betrachten. Eine sinnvolle Anwendung ist hier zum Beispiel ein Bewässerungs-Netzwerk, dessen Knoten Pumpstationen mit einer beschränkten Kapazität sind. Formal führen wir nun eine weitere Abbildung $d : V \to \mathbb{R}_0^+$ ein. Zusätzlich gelte nun für den Fluss f neben dem Kirschhoffschen Gesetz noch die Bedingung

$$\sum_{e \in \delta^+(v)} f(e) \leq d(v) \quad \text{für } v \neq s, t.$$

Reduzieren Sie diese spezielle Variante eines Flussproblemes auf die Ihnen bekannte Definition. Geben Sie also ein Verfahren / eine Vorschrift an, wie man dieses Flussproblem in ein normales Flussproblem überführen kann.

(5 Punkte)

[1] Anmerkung: $s = \sum_{k=0}^{\infty} \sigma^k = \frac{1}{1-\sigma}$ ist der Wert der geometrischen Reihe der σ.

Programmieraufgabe 1. (Präsenzübung: Punktweiser Fluss in einem Netzwerk)

Gegeben sei ein gerichteter Graph im CSR-Format, der ein Netzwerk darstellt. Hierbei repräsentieren die Kantengewichtedie Kapazitäten. Überlegen Sie sich, wie Sie mit Hilfe der Matrix-Vektor Multiplikation den Fluss in jedem Knoten bestimmen können. Konstruieren Sie sich einen gerichteten Hyperwürfel indem Sie die Kanten entlang der Koordinatenachsen richten, das heisst die Kante von Punkten $x = (\star, \dots, \star, 0, \star, \dots, \star)$ nach Punkt $y = (\star, \dots, \star, 1, \star, \dots, \star)$ existiert, jedoch nicht die Kante (y, x). Die Kapazitäten auf den Kanten sollen zufällig aus $(0, 1)$ gewählt werden. Bestimmen Sie den Fluss an jedem Knoten des Hyperwüerfels. Welcher Punkt entspricht in dem Graphen der Quelle und welcher der Senke?

Die Präsenzübung wird in den Programmiertutorien besprochen, die in der Woche vom 11.01.2016-15.01.2016 stattfinden.

Programmieraufgabe 2. (Edmonds-Karp Algorithmus)

In dieser Programmieraufgabe berechnen Sie einen Netzwerkfluß auf der Kugeloberfläche vom Nordpol zum Südpol. Kontruieren Sie dazu einen gerichten Graphen auf der Kugeloberfläche. Verändern Sie dabei Ihre Implementierung der Kugel im CSR-Format wie folgt:

1. Alle Kanten entlang eines Längengrades werden von Norden nach Süden gerichtet und erhalten eine zufällige Kapazität zwischen 0 und 1. Die entsprechenden Rückwärtskanten von Süden nach Norden, die notwendig sind für den Komplementgraphen, werden mit dem Gewicht 0 versehen.
2. Kanten entlang von Breitengeraden werden zufällig nach Westen oder Osten gerichtet und auch mit einem zufälligen Gewicht zwischen 0 und 1 versehen. Versehen Sie wiederum die entsprechenden Rückwärtskanten mit dem Gewicht 0.

Bestimmen Sie anschließend den maximalen Fluss vom Nord- bis zum Südpol in dem so entstandenen Graphen mit Hilfe der Funktion `EdmondsKarp` aus `NetworkFlow.cpp`. Testen Sie die Laufzeit Ihres Programms für $n = 20, 40, 60, \dots, 200$.

(8 Punkte)

Die Programmieraufgabe wird in den Wochen vom 25.01-05.02.2016 im Cip-Pool Endenicher Allee oder im Cip-Pool Wegelerstraße abgegeben/vorgestellt. In den Wochen vom 18.01-29.01.2016 werden in den Cip-Pools Listen für die Abgabe aushängen.