Dr. Arne Schmidt
Christian Rieck

**Abgabe:** 04. Juli 2021
**Rückgabe:** 09. Juli 2021

# Übungsblatt 4

**Bitte schickt eure Lösungen in einer einzigen PDF Datei (Name der Datei und Betreff: „blatt[nr]_[name]_[matrikelnr]") per Mail an euren jeweiligen Tutor. Bitte beachtet auch den Hausaufgaben-Merkzettel[1]!**

---

**Hausaufgabe 1:** **(5 Punkte)**

Wende auf den in Abbildung 1 abgebildeten Graphen $G$ den Algorithmus von Stoer-Wagner an. Gib dabei folgende Punkte an:

- Bei jedem Durchlauf die Knoten in der Reihenfolge wie sie in $X$ aufgenommen werden.
- Den gefundenen Schnitt.
- Den neuen Graphen nach Verschmelzen von zwei Knoten.

Stehen zu einem Zeitpunkt mehrere Knoten zur Auswahl, fahre mit dem Knoten mit dem kleinsten Index fort. Dabei erhalten verschmolzene Knoten den Index des Knotens mit dem kleinsten Index aller darin enthaltenen Knoten.

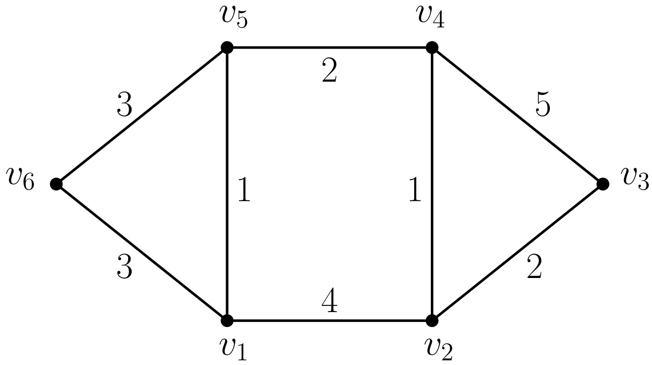


**Abbildung 1:** Abbildung des Graphen $G$.

**Hausaufgabe 2:** **(6 Punkte)**

Im Beweis zu Satz 4.12 wurde eine Menge von Pfaden und Kreisen konstruiert.
Zeige, dass diese Menge die im Satz 4.12 aufgelisteten Eigenschaften erfüllt:

a) $f(e) = \sum_{\substack{p \in \mathcal{P} \cup \mathcal{C} \\ e \in p}} w(p)$,

b) $\text{Wert}(f) = \sum_{p \in \mathcal{P}} w(p)$ und

c) $|\mathcal{P}| + |\mathcal{C}| \leq |E|$.

[1] `https://www.ibr.cs.tu-bs.de/alg/Merkzettel/homework-booklet.pdf`

**Hausaufgabe 3:** **(3+3+3 Punkte)**
Sei $D = (V, E)$ ein Digraph und seien $s, t \in V$. Wir nennen zwei $s$-$t$-Pfade $P$ und $P'$ in $D$ *knoten-disjunkt*, wenn sie nur die Knoten $s$ und $t$ gemeinsam haben, und *kanten-disjunkt*, wenn sie keine gemeinsame Kante besitzen.

a) Gib einen Algorithmus an, der die *maximale* Anzahl gerichteter, *kanten*-disjunkter $s$-$t$-Pfade in $D$ bestimmt.

b) Gib einen Algorithmus an, der die *minimale* Anzahl an Kanten bestimmt, die entfernt werden müssen, sodass es keinen $s$-$t$-Pfad mehr in $D$ gibt.

c) Gib einen Algorithmus an, der die *maximale* Anzahl gerichteter, *knoten*-disjunkter $s$-$t$-Pfade in $D$ bestimmt.

Beweise jeweils die Korrektheit deiner Algorithmen.

**Präsenzaufgabe:**
Seien $G = (V_G, E_G)$ und $K = (V_K, E_K)$ bipartite Graphen. Eine Packung von $K$ in $G$ ist eine Partition von $G$ in Teilgraphen $G_1, \ldots, G_\ell$, sodass (1) $G_i$ isomorph zu $K$ für alle $1 \leq i \leq \ell$ ist und (2) $V_{G_i} \cap V_{G_j} = \emptyset$ für $1 \leq i < j \leq \ell$ ist (d.h. sie überlappen sich nicht; sind also *unabhängig* in $G$).

Die Aufgabe ist nun, möglichst viele Kopien von $K$ in $G$ unterzubringen, d.h. $\ell$ soll maximiert werden. Sei dazu $\ell_{opt}$ die maximale Anzahl an Kopien von $K$, die in $G$ untergebracht werden können. Kann ein Algorithmus $\mathcal{A}$ garantieren, dass mindestens $c \cdot \ell_{opt}$ Kopien für $c \leq 1$ in $G$ platziert werden können, dann ist $\mathcal{A}$ ein $c$-approximierender Algorithmus.

Betrachte folgenden Algorithmus: Solange $K$ unabhängig in $G$ platziert werden kann, platziere $K$ in $G$.

a) Zeige, dass der gegebene Greedy-Algorithmus ein 1/2-approximierender Algorithmus ist, wenn $K := K_{1,1}$.

b) Zeige, dass der gegebene Greedy-Algorithmus ein 1/3-approximierender Algorithmus ist, wenn $K := K_{1,2}$.