Algorithmische Mathematik:

Graphen & Anwendungen

Frühlingssemester 2018
P. Zaspel und I. Kalmykov

Universität Basel

Wiederholungsblatt

Aufgabe 1. (Algorithmen und ihre Korrektheit)

a) Was ist der *Aufwand* bzw. die *(Laufzeit-)Komplexität* eines Algorithmus?

b) Es sei folgender Algorithmus gegeben:

> **Eingabe** $G = (V, E, \gamma)$ gerichtet, gewichtet. $|V| = n$.
>
> **Ausgabe** $[\ell_{ij}]_{i,j=1}^n$, $[p_{ij}]_{i,j=1}^n$ Matrizen, $\ell_{ij} \in \mathbb{R}, p_{ij} \in V$
>
> 1. setze $\ell_{ij} := \gamma((v_i, v_j)) \,\forall (v_i, v_j) \in E$
> 2. setze $\ell_{ij} := \infty \,\forall (v_i, v_j) \in (V \times V) \setminus E$ mit $v_i \neq v_j$
> 3. setze $\ell_{ii} := 0 \,\forall i \in \{1, \ldots n\}$
> 4. setze $p_{ij} := v_i \,\forall v_i \in V, j \in \{1, \ldots, n\}$
> 5. für alle $k = 1, \ldots, n$:
>   für alle $i = 1, \ldots, n$, so dass $i \neq k$:
>   für alle $j = 1, \ldots, n$ so dass $j \neq k$:
>   falls $\ell_{ij} > \ell_{ik} + \ell_{kj}$, dann
>   setze $\ell_{ij} := \ell_{ik} + \ell_{kj}$
>   setze $p_{ij} := p_{kj}$

Bitte beachten Sie zunächst, dass die Eingabe- und Ausgabe-Größen nicht vollständig spezifiziert sind.

(a) Um welchen Ihnen aus der Vorlesung bekannten Algorithmus handelt es sich?

(b) Geben Sie die noch fehlenden Eingabe- und Ausgabe-Spezifikationen, wie sie in der Vorlesung diskutiert wurden, an.

Aufgabe 2. (Grundbegriffe zu Graphen)

a) Definieren Sie den Begriff eines *gerichteten Graphen.*

b) Gegeben sei der folgende gewichtete Graph G in Abbildung 1. Geben Sie diesen formal als 3–Tupel $G = (V, E, \gamma)$ mit den entsprechenden Mengen V, E und der Abbildung γ an.

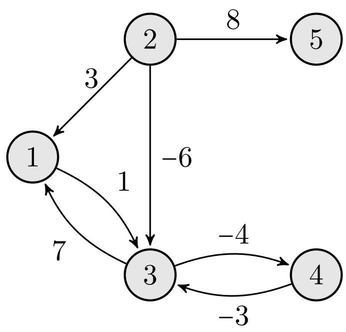

Abbildung 1: Gewichteter Graph für die Aufgabe 2 b)

c) Sei $G = (V, E)$ ein einfacher ungerichteter Graph. Man zeige, dass mindestens einer der Graphen G und $\bar{G}$ zusammenhängend ist, wobei $\bar{G} := (V, \bar{E})$ mit

$$\bar{E} := \{X \subseteq V \mid |X| = 2\} \setminus E$$

das Komplement von G ist.

Aufgabe 3. (Graphenstruktur)

a) Sei $G = (V, E)$ ungerichtet und $|V| > 1$. Zeigen Sie: G hat zwei Knoten vom selben Grad.

b) Wir definieren $\kappa(G)$ als die Anzahl der Zusammenhangskomponenten eines Graphen G. Sei $G = (V, E)$ ein Graph und $e \in E$ beliebig, aber fest. $G' = (V, E \setminus \{e\})$ sei der Graph, der aus G durch entfernen der Kante e entsteht. Zeigen Sie:

$$\kappa(G) \leq \kappa(G') \leq \kappa(G) + 1.$$

Aufgabe 4. (Graphendurchmusterung)

Man führe die Tiefensuche auf dem Graphen in Abbildung 2 mit Startknoten v_1 durch, indem man den entstehenden DFS-Baum und die Durchlaufreihenfolge der Knoten angibt. Beachten Sie dabei, dass die Durchlaufreihenfolge durch die angegebenen Adjazenzlisten eindeutig wird.

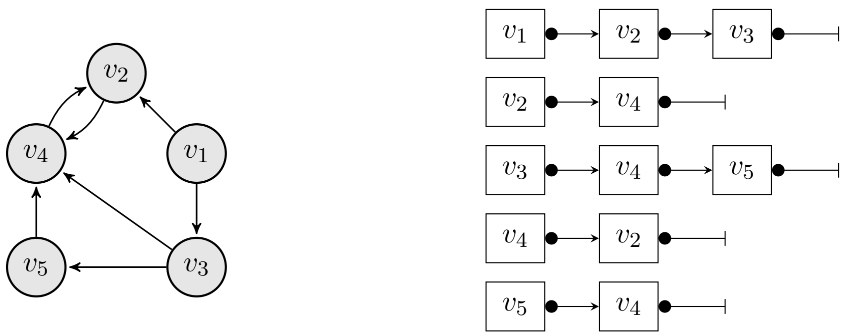

Abbildung 2: Graph für Aufgabe 4

Aufgabe 5. (Kürzeste Wege)

a) Definieren Sie die Weglänge für einen Weg $\pi = v_0, v_1, \ldots, v_r$ bzgl. einer Kostenfunktion γ in einem gewichteten Graphen.

b) Wir betrachten den folgenden gewichteten, gerichteten Graphen in Abbildung 3.

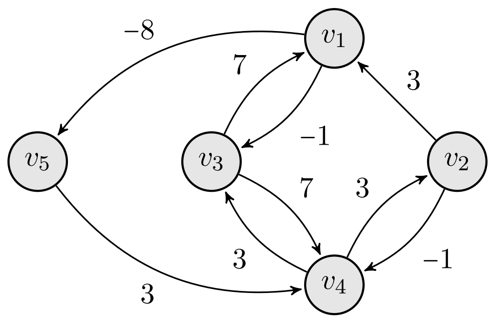

Abbildung 3: Gewichteter Graph für Aufgabe 5 b)

Man berechne mit dem Moore–Bellman–Ford–Algorithmus die kürzesten Weglängen zu allen Knoten ausgehend von Startknoten v_1. Verwenden Sie hierzu die aus der Vorlesung und Übung bekannte tabellarische Darstellung. Schließlich geben Sie den kürzesten Weg von v_1 nach v_3 an.

Aufgabe 6. (Netzwerke)

a) Man definiere den Begriff *Netzwerk*.

b) Beantworten Sie folgende Fragen:

(i) Welche Aufgabe löst der Edmonds–Karp–Algorithmus?

(ii) Welche asymptotische Laufzeit hat der Edmonds–Karp–Algorithmus?

c) Wir betrachten das folgende Netzwerk $N = (G, c, s, t)$ in Abbildung 4.

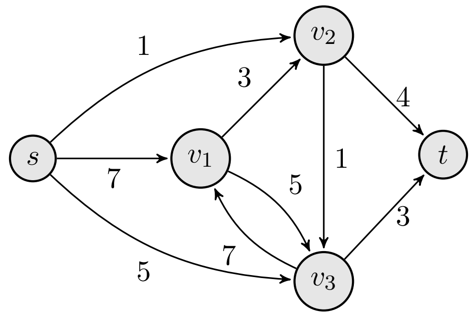

Abbildung 4: Netzwerk für Aufgabe 6 c)

Man berechne mit dem Edmond–Karps–Algorithmus den Wert des maximalen Flusses in obigem Netzwerk. Verwenden Sie hierzu das aus der Vorlesung und Übung bekannte vorgehen. Geben Sie ebenfalls den Wert des minimalen Schnittes (S, T) in N an.

Aufgabe 7. (Programmieraufgabe)

Schreiben Sie einen Algorithmus in Pseudo–Code oder als Matlab/Python/C/C++ –Code, der folgende Ein- und Ausgabespezifikation erfüllt:

Eingabe: $G = (V, E, \gamma)$ gewichteter und gerichteter Graph mit $\gamma : E \to \mathbb{R}$ konservativ.

Ausgabe: Knotenpaar $v_i, v_j \in V$, $v_i \neq v_j$ mit minimaler kürzester Weglänge $\delta(v_i, v_j)$ (minimal bzgl. aller Knotenpaare, so dass $v_i \neq v_j$).

(Hinweis: Paar v_i, v_j ist nicht notwendigerweise eindeutig. Es genügt ein solches Paar zu finden.)