Algorithmische Mathematik:

Graphen & Anwendungen

Frühlingssemester 2018
P. Zaspel und I. Kalmykov
Universität Basel

Übungsblatt 6.

zu bearbeiten bis **Dienstag, 10.4.2018, 14:00 Uhr**.

Aufgabe 1. (Dijkstra)

Gegeben sei der gerichtete Graph $G = (V, E)$ aus Abbildung 1 mit den zugehörigen Adjazenzlisten.

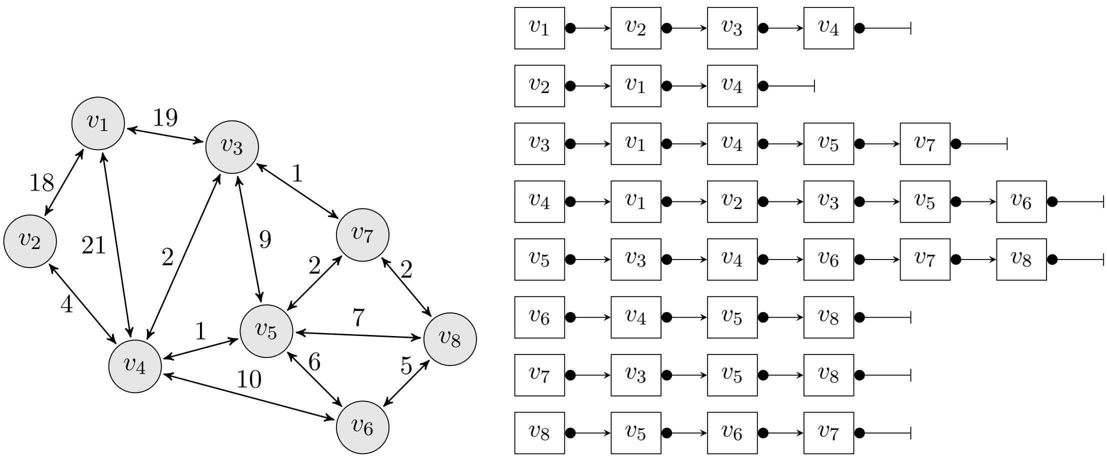

Abbildung 1: Graph G und Adjazenzliste.

a) Berechnen Sie mittels des Dijkstra–Algorithmus schrittweise die Abstände von v_1 zu allen anderen Knoten. Verwenden Sie dabei die tabellarische Darstellung aus der Vorlesung.

b) Geben Sie einen kürzesten Weg von v_1 nach v_8 an.

(4 Punkte)

Aufgabe 2. (Azyklischer Graph)

Gegeben sei ein ungerichteter azyklischer Graph $G = (E, V)$ mit einer Gewichtsfunktion $\gamma \colon E \to \mathbb{R}$.

a) Geben Sie einen Algorithmus an, um für $s, t \in V$ mit $t \in \mathrm{post}^\star(s)$ einen kürzesten s-t-Weg in G in linearer Zeit $\mathcal{O}(|V| + |E|)$ zu finden.

b) Zeigen Sie, dass die Laufzeit tatsächlich linear ist, d.h. $\mathcal{O}(|V| + |E|)$.

(4 Punkte)

Aufgabe 3. (Breitmöglichste Wege)

Gegeben sei ein gewichteter gerichteter Graph $G = (V, E, \gamma)$ mit $\gamma\colon E \to \mathbb{R}_+$. Für einen Weg π_{st} von s nach t, wobei $s, t \in V$, ist die **Breite** des Weges gegeben durch die Kante mit dem kleinsten Gewicht:

$$\mathrm{width}(\pi_{st}) := \min_{e \in E(\pi_{st})} (\gamma(e)).$$

Dabei bezeichnet $E(\pi_{st})$ die Menge der Kanten im Pfad π_{st}. Gesucht ist ein Algorithmus, der zu gegebenem $s \in V$ den breitesten Weg π_{opt} von s nach t, $\pi_{\mathrm{opt}} = \mathrm{argmax}_{\pi_{st}}(\mathrm{width}(\pi_{st}))$ für alle $t \in V$ findet. Ändern sie hierzu den Algorithmus von Dijkstra entsprechend.

(4 Punkte)

Aufgabe 4. (Implementierung von Dijkstra)

Implementieren Sie den Dijkstra-Algorithmus zur Berechnung der Einzelquelle-kürzeste-Wege für gerichtete Graphen mit nicht-negative Gewichten.

Algorithmus (Dijkstra)

Input: gerichteter, gewichteter Graph $G = (V, E, \gamma)$ mit nichtnegativen Gewichten, Startknoten $s \in V$

Output: Länge eines kürzesten s-v-Weges $l(v)$ für alle Knoten $v \in \mathrm{post}^\star(s)$, Vorgänger $p(v)$ von $v \in \mathrm{post}^\star(s)$ auf dem kürzesten s-v-Weg, für $v \notin \mathrm{post}^\star(s)$ ist $l(v) = \infty$ und $p(v)$ ist undefiniert.

$l(s) \leftarrow 0$
for $v \in V \setminus \{s\}$ **do**
  $l(v) \leftarrow \infty$
end for
$R \leftarrow \emptyset$
while $V \setminus R \neq \emptyset$ **do**
  wähle $u \in V \setminus R$ mit $l(u)$ minimal
  $R \leftarrow R \cup \{u\}$
  for $v \in (V \setminus R) \cap \mathrm{post}(u)$ **do**
    if $l(v) > l(u) + \gamma((u, v))$ **then**
      $l(v) \leftarrow l(u) + \gamma((u, v))$
      $p(v) \leftarrow u$
    end if
  end for
end while
return l, p

Testen Sie die Implementierung an dem Graphen aus Aufgabe 1, welcher mit der Adjazenzliste aus Abbildung 1 abgespeichert ist. Wählen Sie v_1 als Startknoten.

Hinweis. In der Zuweisung $l(v) \leftarrow \infty$ kann ∞ durch einen hinreichend großen Wert ersetzt werden, zum Bsp. $(\max_{e \in E} \gamma(e)) \cdot |E| + 1$. Wichtig ist, dass der Wert echt größer als die Länge von einem beliebigen kürzesten Weg ist. Der undefinierte Wert von $p(v)$ kann mit $p(v) = -1$ realisiert werden.

(4 Punkte)