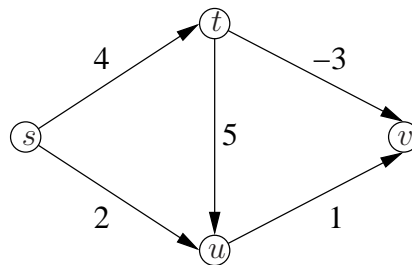


Theoretische Informatik I

8. Übung

Geben Sie die Lösung der Aufgabe 1 bitte bis zum 09.12.2013 9:00 Uhr ab. (Briefkasten vorm Raum 1/266 oder per eMail an fal@informatik.tu-chemnitz.de, *Betreff: TI1 Hausaufgaben*)

1. Aufgabe: Wir betrachten den folgenden gerichteten Graphen $G = (V, E)$ mit Kantengewichten.



Die Knoten sind wie folgt nummeriert:

$$s \rightarrow 1, \quad t \rightarrow 2, \quad u \rightarrow 3, \quad v \rightarrow 4$$

- Bestimmen Sie die kürzesten Wege vom Knoten s aus mit *Dijkstras Algorithmus*. Geben Sie auch die Zwischenstritte an. Was fällt Ihnen bezüglich der ermittelten Weglängen auf?
- Kann man das Problem *im Allgemeinen beheben*, indem man zu *allen* Kanten den Wert $-x$ addiert? Hier sieht es zunächst so aus, als würde das funktionieren. Warum geht das trotzdem *nicht*? (Geben Sie ein Beispiel oder eine allgemeine Begründung an.)
(Der Wert x bezeichnet hier das kleinste Kantengewicht in G . Durch die Addition von $-x$ werden also alle Kantengewichte ≥ 0 .)
- Ermitteln Sie die Länge der kürzesten Wege in G mit dem *Floyd-Warshall-Algorithmus*. Geben Sie den Inhalt der Matrix *am Anfang* und *nach jedem Durchlauf* der äußeren Schleife an.
- Zusatzaufgabe:* Wie muss der Floyd-Warshall-Algorithmus modifiziert werden, um nicht nur die Länge sondern auch die konkreten Wege zu erhalten. Wie kann man die Wege dann ausgeben?

Hinweis: Es muss für jedes Paar (u, v) in der Matrix nur eine zusätzliche Information gespeichert werden. Die Ausgabe erfolgt dann rekursiv.

2. Aufgabe: Modifizieren Sie *Dijkstras Algorithmus*, so dass neben der Länge auch die Anzahl aller (einfachen) kürzesten Wege berechnet werden.

3. Aufgabe: Betrachten Sie den Fall, dass bei der Ausführung des *Floyd-Warshall-Algorithmus* die äußere Schleife für $k = 3$ gerade durchlaufen wurde. Welche denkbaren Wege zwischen Knoten zwei beliebigen, aber festen Knoten u und v sind zu diesem Zeitpunkt bereits untersucht worden?

4. Aufgabe: Was passiert beim *Floyd-Warshall-Algorithmus*, wenn der Graph einen Kreis mit *negativer Länge* enthält?