



# Datenstrukturen und Algorithmen (SS 2013)

## Übungsblatt 10

Abgabe: Montag, **08.07.2013**, 14:00 Uhr

- Die Übungen sollen in Gruppen von zwei bis drei Personen bearbeitet werden.
- Schreiben Sie die Namen jedes Gruppenmitglieds sowie alle Matrikelnummern auf die abgegebenen Lösungen.
- Schreiben Sie die Namen jedes Gruppenmitglieds sowie alle Matrikelnummern auch in die Quellcode-Dateien.
- Geben Sie Ihre Lösungen am **Anfang** der Globalübung, montags, 14:00 Uhr, ab.
- Schicken Sie den jeweiligen Quellcode bitte per **E-Mail** direkt an Ihre/n Tutor/in.
- Geben Sie außerdem den ausgedruckten Quellcode zusammen mit den schriftlichen Lösungen ab.
- Zu spät abgegebene Lösungen werden nicht bewertet.
- Sofern nicht anders gefordert, müssen alle Lösungen und Zwischenschritte kommentiert werden.



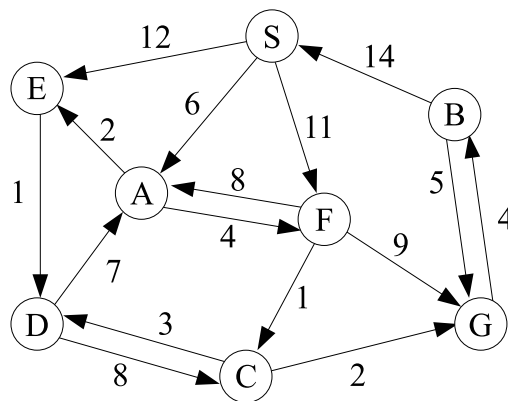
**Aufgabe 1** (*Maximaler Abstand in Graphen* [6 Punkte])

Sei  $G = (V, E)$  ein Baum mit Gewichtsfunktion  $w : E \rightarrow \mathbb{R}^+$ . Gesucht sind zwei Knoten  $u, v \in V$ , deren Abstand maximal ist (d.h. die Kosten des kürzesten Pfades von  $u$  nach  $v$  sollen maximal sein). Beschreiben und begründen Sie einen Algorithmus, der die beiden Knoten  $u$  und  $v$  und den Abstand von  $u$  und  $v$  in Laufzeit  $O(|V|)$  bestimmt.

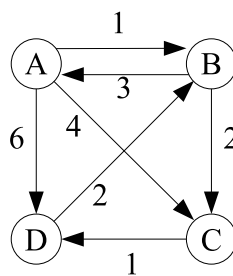


**Aufgabe 2** (*Kürzeste Pfade* [4 Punkte])

- Wenden Sie Dijkstras Algorithmus auf den folgenden Graphen  $G$  mit Startknoten  $S$  an. Geben Sie die Werte aller oberen Schranken nach jedem Durchlauf der Hauptschleife von Dijkstras Algorithmus in Form einer Tabelle an. Die Tabelle enthält also eine Spalte für jeden Knoten  $v \in G$ . Pro Iteration soll eine Zeile hinzugefügt werden, die die aktuellen oberen Schranken  $\delta(S, v)$  enthält. Markieren Sie dabei in jeder Zeile den Knoten, den Sie auswählen. [2 Punkte]



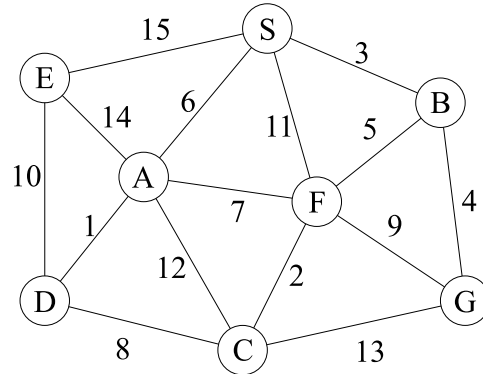
- Wenden Sie den Algorithmus von Floyd-Warshall auf den unten angegebenen Graphen an, um die Längen aller kürzesten Pfade zu berechnen. Geben Sie die Distanzmatrix nach jedem Durchlauf der Hauptschleife an. [2 Punkte]





**Aufgabe 3** (*Minimale Spannbäume* [6 Punkte])

Gegeben sei folgender Graph:



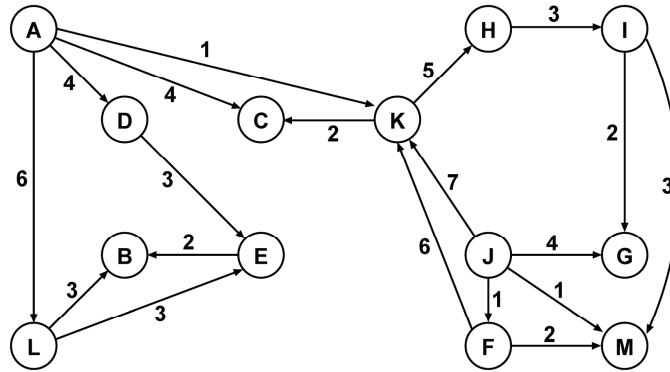
1. Wenden Sie den Algorithmus von Prim an, um einen minimal spannenden Baum des Graphen zu berechnen. Der Startknoten sei  $S$ . [3 Punkte]
2. Wenden Sie den Algorithmus von Kruskal an, um einen minimal spannenden Baum des Graphen zu berechnen. [3 Punkte]

Geben Sie in die Kanten in der Reihenfolge an, in der sie zur jeweiligen Lösungsmenge hinzugefügt werden. Da alle Kantengewichte verschieden sind, genügt es, jede Kante durch ihr Gewicht zu identifizieren.



**Aufgabe 4** (*Topologisches Sortieren* [4 Punkte])

Gegeben sei der folgende gerichtete Graph  $G$ :



1. Bestimmen Sie für  $G$  eine topologische Sortierung und geben Sie die Knoten als sortierte Folge an. Die Kantengewichte können hierbei vernachlässigt werden. [3 Punkte]
2. Erweitern Sie  $G$  um eine beliebige Kante (ohne Gewicht), so dass keine topologische Sortierung für den modifizierten Graphen mehr existiert. [1 Punkt]