

Dr. Christiane Schmidt  
Florian Maurer  
Arne Schmidt

## Netzwerkalgorithmen Übung 1 vom 28.04.2014

Abgabe der Lösungen bis Mittwoch, den 14.05.2014, bis 13:00 Uhr in der  
Abteilung *Algorithmik*.

Bitte die Blätter vorne deutlich mit eigenem Namen und Gruppennummer versehen!

### **Aufgabe 1 (Algorithmische Grundprinzipien - Divide and Conquer):**

Bereite einen Kurzvortrag zum Thema "Divide and Conquer" vor - d.h. Du solltest in 5-15 Minuten in der Lage sein, deinen Kommilitonen zu erläutern, was das Grundprinzip ist, ein Beispiel geben und auch erläutern können, warum dieses algorithmische Prinzip dann leider doch nicht für jedes Problem die optimale Lösung liefert. Eine mögliche Quelle für diese Aufgabe kann z.B. das entsprechende Kapitel im Buch von Vazirani sein: <http://www.cs.berkeley.edu/~vazirani/algorithms/chap2.pdf>.

**Erläuterung der Punktevergabe für diese Aufgabe:** Wir nennen das mal Vertrauensaufgabe - Ihr tragt Euch zu Beginn der kleinen Übung in eine Liste ein, womit Ihr sagt "Ja, ich habe diese Aufgabe bearbeitet, möchte also gerne die 15 Punkte, d.h. auch, wenn ich ausgewählt werde, kann ich problemlos den Kurzvortrag halten." - Florian und Arne wählen dann zufällig jemanden aus der Liste, und wenn der-/diejenige dann keinen Kurzvortrag hat, gibt es logischerweise keine Punkte, und ein Nächster wird ausgewählt - alles klar?

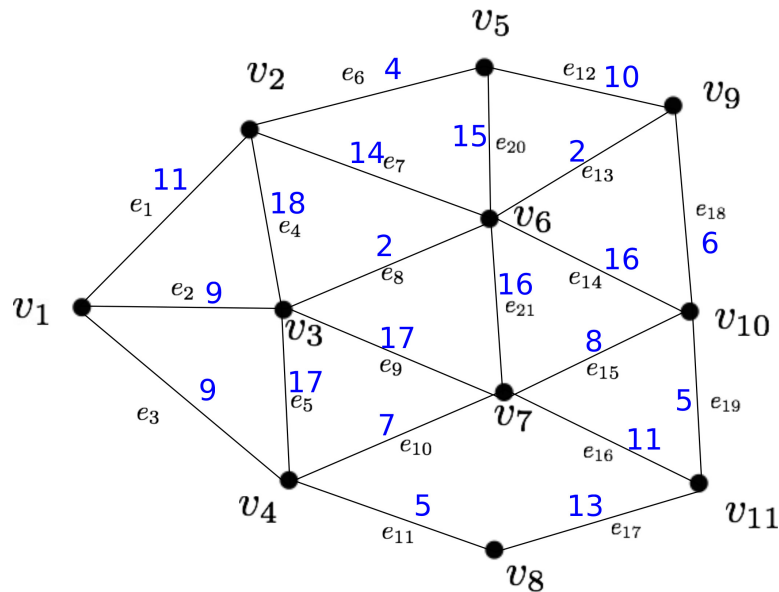
(15 Punkte)

### **Aufgabe 2 (Zusammenhang):**

- (a) Sei  $G$  ein Graph mit  $n$  Knoten. Angenommen, jeder Knoten von  $G$  hat mindestens Grad  $(n-1)/2$ . Zeige, dass  $G$  zusammenhängend sein muss.
- (b) Zeige: Ein Graph  $G$  ist zusammenhängend genau dann, wenn für jede Zerlegung der Knotenmenge  $V(G) = V_1 \cup V_2$  (d.h.  $V_1 \cap V_2 = \emptyset$ ) eine Kante  $e = \{v, w\}$  mit  $v \in V_1$  und  $w \in V_2$  existiert.
- (c) Zeige: Wenn  $G$  nicht zusammenhängend ist, dann ist der komplementäre Graph  $\bar{G}$  zusammenhängend. (Für einen Graph  $G = (V, E)$  enthält der komplementäre Graph alle zum vollständigen Graphen auf  $V$  fehlenden Kanten.)

- (d) Zeige: Ein zusammenhängender Graph mit  $n$  Knoten hat mindestens  $n-1$  Kanten.  
(5+5+5+5 Punkte)

**Aufgabe 3 (Algorithmus von Kruskal):**



Bestimme mit Hilfe des Algorithmus von Kruskal einen minimalen aufspannenden Baum. Gib dazu die Kanten in der Reihenfolge an in der sie in den Baum aufgenommen werden und zeichne die gefundene Gesamtlösung. Kommen in einem Schritt mehrere Kanten in Frage, wähle die mit dem kleinsten Kantenindex.

Um die Korrektur zu vereinfachen, gehen wir davon aus, dass wir Kanten immer in der Form  $e = (v_i, v_j)$  mit  $i < j$  schreiben.

**WICHTIG:** Damit der Algorithmus von Kruskal eine Laufzeit von  $O(m \log n)$  erreicht, kann man die in der Vorlesung vorgestellte Datenstruktur verwenden. Gib jeweils nach dem Einfügen einer Kante den Zustand der Datenstruktur an.

(Hinweis: Kommen beim Einfügen einer Kante in die Datenstruktur zwei Möglichkeiten in Frage, wähle die Kante, so dass sie vom Knoten mit dem kleineren Index zum Knoten mit dem größeren Index verläuft.)

(15 Punkte)

**Einschub - Gerichtete Graphen:** Für einen gerichteten Graphen  $D = (V, A)$  ist  $G = (V, E)$  mit  $E = \{\{v, w\} | (v, w) \in A\}$  der zu  $D$  gehörende ungerichtete Graph. Ein gerichteter Graph ist zusammenhängend, wenn der ungerichtete Graph zusammenhängend ist.

Ein gerichteter Graph ist ein *Branching*, wenn der zugrundeliegende ungerichtete Graph ein Wald ist, und jeder Knoten nur eine eingehende Kante hat. Eine *Arboreszenz* ist

ein zusammenhängendes Branching. Der eindeutige Knoten in einer Arboreszenz mit Eingrad 0 ist die *Wurzel*. Ein gerichteter Graph ist *azyklisch*, wenn er keinen gerichteten Kreis enthält.

**Aufgabe 4 (Kürzeste  $r$ -Arboreszenzen):** Gegeben sei ein gerichteter Graph  $D = (V, A)$ , ein Knoten  $r$ , und eine Längenfunktion  $\ell : A \rightarrow \mathbb{Q}_+$ .

Wir suchen eine kürzeste  $r$ -Arboreszenz (also eine in  $r$  verwurzelte Arboreszenz).

Der Greedy-Algorithmus für dieses Problem lautet folgendermaßen:

Starte bei  $r$  und erweitere iterativ eine  $r$ -Arboreszenz der Teilmenge  $U$  von  $V$  durch die kürzeste Kante, die aus  $U$  herausgeht.

Liefert dieser Algorithmus eine kürzeste  $r$ -Arboreszenz? Beweise Deine Behauptung.

(10 Punkte)