Fachbereich IV - Abteilung Informatik
Algorithmen und Datenstrukturen
Universität Trier

Prof. S. Näher Tel.: 0651 201-3275

9. Übung zur Vorlesung:

# Algorithmen und Datenstrukturen

Sommersemester 2006 13. Juli 2006

---

**Aufgabe 9.1:** (Punkte 6)

Modifizieren Sie den in der Vorlesung behandelten Algorithmus `explorefrom(s)` zum Durchmustern eines gerichteten Graphen so, dass er für jeden Knoten $v$ die Länge (d. h. Anzahl der Kanten) eines kürzesten Pfades von $s$ nach $v$ berechnet.

**Aufgabe 9.2:** (Punkte 8)

Ein Graph $G = (V, E)$ heißt *ungerichtet*, wenn für jede Kante $(v, w) \in E$ auch $(w, v) \in E$ gilt. Im Allgemeinen zerfällt ein ungerichteter Graph in mehrere nicht miteinander verbundene Teile (Zusammenhangskomponenten). Verwenden und erweitern Sie die explorefrom-Funktion, um diese Teile zu berechnen. Genauer, wenn $G$ aus $\ell$ Zusammenhangskomponenten $K_1, \ldots, K_\ell$ besteht, dann berechnen Sie für jeden Knoten $v$ eine Zahl $num[v] \in \{ 1, \ldots, \ell \}$ mit $num[v] = i$ genau dann, wenn $v \in K_i$. *Hinweis:* Rufen Sie im Hauptprogramm `explorefrom(v)` für jeden noch nicht zuvor besuchten Knoten $v$ auf und erweitern Sie die Funktion so, dass die gesuchten Knotennummern zugewiesen werden.

**Aufgabe 9.3:** (Punkte 6)

Sei $G = (V, E)$ ein azyklischer gerichteter Graph und $compnum : V \longrightarrow \{1, \ldots, n\}$ eine Nummerierung der Knoten, die der Abschlussreihenfolge (*completion number*) der rekursiven DFS-Aufrufe entspricht. Zeigen Sie, dass die Abbildung $ord : V \longrightarrow \{1, \ldots, n\}$ mit $ord(v) = n + 1 - compnum(v)$ eine topologische Sortierung von $G$ ist.