INSTITUT FÜR INFORMATIK V
UNIVERSITÄT BONN

Prof. Dr. Norbert Blum
Matthias Kretschmer

Römerstraße 164
53117 Bonn

19. November 2007

Informatik III
Algorithmen und Berechnungskomplexität I
WS 2007/08
Übungen – Blatt 5

Abgabe: 26.11.2007 bis 11Uhr vor HS D

Aufgabe 1 (10 Punkte)**.** Sei $G = (V, E)$ ein gerichteter Graph. Zeigen Sie, dass bei der Tiefensuche mit Startknoten s jede Kante $(v, w) \in E$ betrachtet wird, falls PUSH(v) während der Tiefensuche durchgeführt wird.

Aufgabe 2 (30 Punkte)**.** Sei $G = (V, E)$ ein gerichteter Graph. Eine topologische Sortierung von G ist eine Abbildung $\sigma : V \to \{1, 2, \ldots, |V|\}$, so dass gilt: $((v, w) \in E) \Rightarrow \sigma(v) < \sigma(w)$.

1. Zeigen Sie: G kreisfrei $\Leftrightarrow$ es existiert eine topologische Sortierung für G.
2. Entwickeln Sie einen Algorithmus, der für einen gerichteten Graphen G eine topologische Sortierung berechnet bzw. eine Fehlermeldung ausgibt, falls keine solche Sortierung für G existiert.

   Geben Sie die Laufzeit Ihres Algorithmus in O-Notation an.

Aufgabe 3 (15 Punkte)**.** Sei $G = (V, E)$ ein gerichteter Graph. Die transitive Hülle von G ist der Graph $G' = (V, E')$ mit $E' = \{(v, w) \mid \text{es existiert ein Pfad von } v \text{ nach } w \text{ in } G\}$.

Geben Sie einen Algorithmus zur Berechnung der transitiven Hülle eines Graphen G an. Bestimmen Sie die Laufzeit.

Aufgabe 4 (20 Punkte)**.** Sei $G = (V, E)$ ein ungerichteter und zusammenhängender Graph. Ein Eulerscher Kreis in G ist eine Folge $(v_1, e_1, v_2, \ldots, e_m, v_1)$ mit $e_i = (v_i, v_{i+1})$ für alle $i : 1 \leq i < m$, $e_m = (v_m, v_1)$, $e_i \neq e_j$ für alle $i \neq j$ und $\{e_1, \ldots, e_m\} = E$. Das heißt, ein Eulerscher Kreis in G ist ein Weg, der alle Kantes des Graphen genau einmal überquert und wieder im Anfangsknoten ankommt.

Zeigen Sie, dass genau dann ein Eulerscher Kreis in G existiert, wenn die Grade aller Knoten gerade sind.

Aufgabe 5 (15 Punkte)**.** Geben Sie einen Pseudocode an, der MERGE(A, p, q, r) mit maximal $r - p$ Vergleichsoperationen realisiert.

Beweisen Sie die Korrektheit und Laufzeit ihres Algorithmus.

□