Algorithmische Mathematik I

Wintersemester 19/20
Prof. Dr. J. Gedicke
Johannes Rentrop und Jannik Schürg

UNIVERSITÄT BONN

Übungsblatt 9.

Abgabedatum: **16.12.2019**

Aufgabe 1. (Topologische Ordnung)

Sei $G = (V, E)$ ein gerichteter Graph. Eine Reihenfolge $v_1, \dots, v_n$ der Knoten von G heißt *topologische Ordnung* falls

$$(v_i, v_j) \in E \Rightarrow i < j.$$

a) Zeigen Sie, dass G eine topologische Ordnung besitzt gdw. G keinen Kreis enthält.

b) Geben Sie einen Algorithmus an, der eine topologische Ordnung in $\mathcal{O}(|V| + |E|)$ berechnet oder entscheidet, dass keine existiert, indem Sie die Tiefensuche (Algorithmus 1 auf diesem Blatt) modifizieren und ggf. als Subroutine verwenden. Zeigen Sie, dass Ihr Algorithmus korrekt ist und die angegebene Laufzeit hat.

c) Der Graph in Abbildung 1 soll Ihnen bei Ihrer Morgenroutine helfen. Wenden Sie Ihren Algorithmus aus b) auf diesen Graphen an, um eine topologische Ordnung zu erzeugen. Geben Sie genügend Zwichenschritte/-ergebnisse an. Einer der Vorlesungsassistenten behauptet, er ziehe die Unterhose nach den Schuhen an. Ist das möglich?

(3+5+2 Punkte)

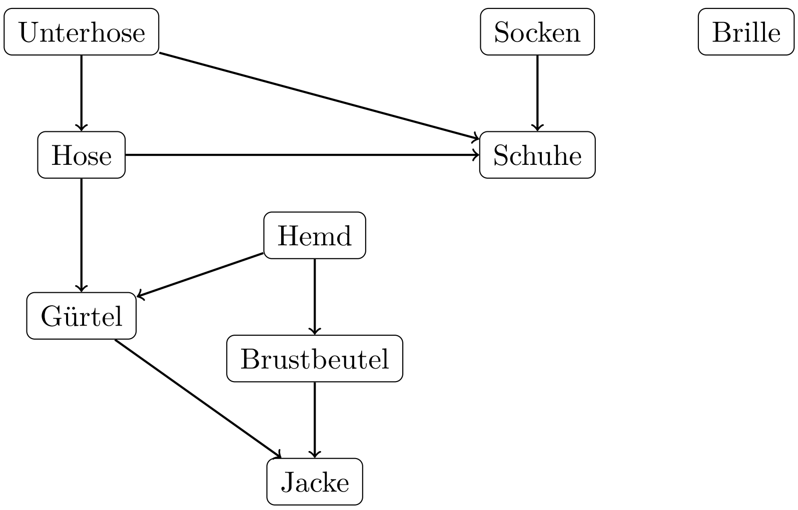

Abbildung 1: Ein Graph, der die Beziehung „muss angezogen werden vor“ für Kleidungsstücke angibt. Dabei soll eine gerichtete Kante (v, w) zwischen zwei Kleidungsstücken bedeuten, dass Kleidungsstück v vor Kleidungsstück w angezogen werden muss.

Aufgabe 2. (Minimum Spanning Tree)

Sei $G = (V, E)$ ein zusammenhängender, ungerichteter Graph mit Kantengewichten $c \colon E \to \mathbb{R}$.

a) Entscheiden Sie und zeigen Sie, wie die folgenden Aussagen zueinander stehen.

i) Es gibt genau einen MST.

ii) Die Funktion c ist injektiv.

b) Zeigen, dass man einen MST erhält, falls aus G sukzessive eine schwerste Kante entfernt wird, die nicht den Zusammenhang zerstört.

(3+3 Punkte)

Aufgabe 3. Sei $G = (V, E)$ ein gerichteter Graph mit Kantengewichten

$$c\colon E \to \{0, 1, \ldots, k\}.$$

Zeigen Sie, dass es einen Algorithmus gibt, der einen Kürzestete-Wege-Baum mit Wurzel s in $\mathcal{O}(kn + m)$ berechnet, wobei $n := |V|$ und $m := |E|$.

(4 Punkte)

Veranstaltungshinweis: Am 16.12.2019 ab 18 c.t. im Lipschitzsaal veranstaltet die Fachschaft Mathematik in Kooperation mit dem HCM ein Treffen für die Teilnehmerinnen und Teilnehmer der Grundvorlesungen. Es soll im Plenum und in Kleingruppen über den Studieneinstieg und die Tutorien gesprochen und diskutiert werden. Alle Beteiligten freuen sich über ein zahlreiches Erscheinen und eine Vielfalt an Rückmeldungen, um auch künftig an den studienbegleitenden Angeboten zu arbeiten. Für ein paar Snacks wird gesorgt.

Algorithmus 1 Tiefensuche

function DFS(Gerichteter Graph $G = (V, E)$, Knoten $r \in V$)
    $R \leftarrow \{r\}$ ▷ Bereits besuchte Knoten
    $Q \leftarrow \{r\}$ ▷ Stack mit Knoten von denen weitergesucht werden muss
    while $Q \neq \emptyset$ **do**
        Wähle das neueste $v \in Q$
        if es gibt $(v, w) \in \delta^+(v)$ mit $w \notin R$ **then**
            $R \leftarrow R \cup \{w\}$
            $Q \leftarrow Q \cup \{w\}$
        else
            $Q \leftarrow Q \setminus \{v\}$

Algorithmus 2 Dijkstras Algorithmus

function DIJKSTRA($G = (V, E)$, $c\colon E \to \mathbb{R}_{\geq 0}$, $s \in V$)
    $R \leftarrow \emptyset$, $Q \leftarrow \{s\}$, $l(s) = 0$
    $l(v) \leftarrow \infty$ for $v \in V \setminus \{s\}$
    while $Q \neq \emptyset$ **do**
        Wähle und entferne ein v aus Q mit $l(v)$ minimal
        $R \leftarrow R \cup \{v\}$
        for all $e = (v, u) \in \delta^+(v)$ mit $l(v) + c(e) < l(u)$ **do**
            $l(u) \leftarrow l(v) + c(e)$
            $p(u) \leftarrow e$
            füge u ein in Q, falls nicht vorhanden
    $T \leftarrow \{p(v) \mid v \in R \setminus \{s\}\}$
    return R, T