Lehrstuhl für Informatik I
— Informationsstrukturen und
wissensbasierte Systeme —
Universität Würzburg

Prof. Dr. H. Noltemeier
Dipl. Math. S.O. Krumke

Würzburg, 18. Juni 1997

Übungen zur Vorlesung
Graphentheoretische Konzepte und Algorithmen (SS97)

1. Übungsblatt

Aufgabe 1: (Speicherung von Graphen)

In dieser Aufgabe sei $G = (V, R, \alpha, \omega)$ ein endlicher gerichteter Graph.

(a) Der *inverse Graph* G^{-1} zu G ist definiert durch $G^{-1} = (V, R^{-1}, \alpha', \omega')$ mit $R^{-1} := \{\, r^{-1} : r \in R \,\}$ wobei $\alpha'(r^{-1}) = \omega(r)$ und $\omega'(r^{-1}) = \alpha(r)$. Der Graph G^{-1} entsteht aus G also dadurch, daß die Richtung aller Pfeile „umgedreht" wird.

Geben Sie sowohl für die Adjazenzmatrixspeicherung als auch für die Adjazenzlistenspeicherung von G effiziente Algorithmen an, um G^{-1} aus G zu berechnen. Bestimmen Sie die Laufzeit Ihrer Algorithmen. (**2+2 Punkte**)

(b) Geben Sie einen Algorithmus an, der in $\mathcal{O}(|V|)$ Zeit feststellt, ob ein einfacher Graph G in Adjazenzmatrixspeicherung eine Ecke v mit $g^+(v) = 0$ und $g^-(v) = |V| - 1$ enthält. (Hinweis: G kann höchstens *eine* solche Ecke v enthalten.) (**3 Punkte**)

Aufgabe 2: (Starker Zusammenhang)

(a) Beweisen Sie, daß die in Definition 3.10 der Vorlesung definierte Relation „stark zusammenhängend ($\sim$)" eine Äquivalenzrelation ist. (**2 Punkte**)

(b) Sei $G = (V, R, \alpha, \omega)$ ein gerichteter endlicher stark zusammenhängender Graph mit $|V| \geq 2$. Zeigen Sie, daß dann $|R| \geq |V|$ gilt. (**2 Punkte**)

Aufgabe 3: (Wege)

Sei $G = (V, R, \alpha, \omega)$ ein gerichteter Graph und w ein Weg in G mit $\alpha(w) = u$ und $\omega(w) = v$. Zeigen Sie, daß es dann einen *elementaren* Weg w' in G mit $\alpha(w') = u$ und $\omega(w') = v$ gibt. (**2 Punkte**)

Aufgabe 4: (Ungerichtete Graphen)

Sei $G = (V, E, \gamma)$ ein ungerichteter endlicher Graph. Mit $k(G)$ bezeichnen wir die *Anzahl der Zusammenhangskomponenten* von G. Sei $e \in E$ eine Kante von G und $G - e := (V, E \setminus \{e\}, \gamma)$ der Graph, den man erhält, wenn man aus G die Kante e entfernt. Beweisen Sie, daß dann die Ungleichung

$$k(G) \leq k(G - e) \leq k(G) + 1$$

gilt. (**4 Punkte**)

Abgabetermin: Mittwoch, 21. Mai 1997, bis 11^{30} Uhr in den dafür vorgesehenen Kästen vor der Teilbibliothek Mathematik/Informatik

Hinweise: Bis zu zwei Übungsteilnehmer dürfen gemeinsam ein Übungsblatt abgeben. Zur Teilnahme an der Klausur sind mindestens 40 Prozent der Punkte in den Übungen während des Semesters notwendig.

Die erste Übung findet am Freitag, den 23 . Mai, statt. Die Gruppeneinteilung entnehmen Sie bitte den Aushängen.

Lösungen:

Aufgabe 1

(a) Für die Adjazenzmatrix des inversen Graphen G^{-1} gilt $A(G^{-1}) = A^T(G)$, wobei wir mit $A^T(G)$ die Transponierte der Matrix $A(G)$ bezeichnen.

Die Transponierte einer Matrix läßt sich durch Vertauschen der Zeilen- und Spalten-Indizes berechnen. Sei $A(G) = (a_{ij})$, dann liefert Algorithmus 1 die Matrix $A(G^{-1}) = (a'_{ij})$.

Algorithmus 1 Berechnung von G^{-1} bei Adjazenzmatrixdarstellung

Input: Ein gerichteter einfacher Graph $G = (V, R)$ in Adjazenzmatrixdarstellung

for $1 \leq i \leq |V|$ **do**
  for $1 \leq j \leq |V|$ **do**
    $a'_{ij} \leftarrow a_{ji}$

Algorithmus 1 besitzt eine Laufzeit von $\Theta(|V|^2)$. Diese Laufzeit ist offenbar Worst-Case optimal, da man zur Erstellung der Adjazenzmatrix des inversen Graphen jeden der $\Theta(|V|^2)$ Einträge der Adjazenzmatrix von G betrachten muß.

Zur Berechnung der Adjazenzliste Adj' des inversen Graphen durchläuft man alle Pfeile des Ursprungsgraphen, d.h. alle Listenelemente seiner Adjazenzliste Adj, und fügt die Pfeile mit vertrauschtem Anfangs- und Endknoten in die Liste des inversen Graphen ein. Dieses Verfahren ist in Algorithmus 2 dargestellt.

Algorithmus 2 Berechnung von G^{-1} bei Adjazenzlistendarstellung

Input: Ein gerichteter einfacher Graph $G = (V, R)$ in Adjazenzlistendarstellung

for all $u \in V$ **do**
  for all $v \in \mathrm{Adj}[u]$ **do**
    Füge den Endkonten u an den Anfang der Liste $\mathrm{Adj}'[v]$ ein

Da das Einfügen an den Anfang einer Liste in konstanter Zeit möglich ist, beträgt die Laufzeit $\Theta(|V| + |R|)$. Auch dieser Algorithmus ist von der Laufzeit her Worst-Case optimal, da die Eingabegröße für einen Graphen in Adjazenzlistendarstellung $\Theta(|V| + |R|)$ ist und man in einem korrekten Algorithmus die Eingabe komplett betrachten muß.

(b) Sei $G = (V, R)$ ein gerichteter einfacher Graph mit $n := |V|$ Ecken und $A(G) = (a_{ij})$ seine Adjazenzmatrix.

Behauptung 1: Es gibt höchstens eine Ecke $v \in V$ mit $g^+(v) = 0$ und $g^-(v) = n - 1$.

Beweis: Da G keine Schlingen enthält, folgt aus $g^-(v) = n-1$, daß für alle $w \neq v$ der Pfeil (w,v) in R enthalten ist. Damit kann außer v keine Ecke Innengrad $n-1$ und Außengrad 0 besitzen. erfüllen. □

Folgerung 2: Sei $i \neq j$. Dann gilt:

- falls $a_{ij} = 1$: die Ecke i kommt nicht in Betracht.
- falls $a_{ij} = 0$: die Ecke j kommt nicht in Betracht.

Algorithmus

In Algorithmus 3 ist ein Verfahren angegeben, daß in $\mathcal{O}(n)$ Zeit testet, ob eine Ecke v mit $g^-(v) = n-1$ und $g^+(v) = 0$ in G existiert.

Algorithmus 3 Reduzierung der Suche auf eine Ecke.

Input: Ein gerichteter einfacher Graph $G = (V, R)$ in Adjazenzmatrixdarstellung mit $n := |V|$ Ecken

Setze $i \leftarrow 2$ (Zeilen-) und $j \leftarrow 1$ (Spalten-Zähler)
while $i \leq n$ **do**
  if $a_{ij} = 0$ **then**
    $j \leftarrow i$
  $i \leftarrow i + 1$

Behauptung 3: Nach Durchlauf des Algorithmus kommt nur die Ecke j als Kandidat in Betracht.

Beweis: Wir zeigen das Ergebnis mit Induktion nach den Schritten des Algorithmus. Sei

$$M(i,j) := \{1, 2, \ldots, i-1\} \setminus \{j\}.$$

Dann gilt: Steht der Algorithmus an Position (i,j), dann können alle Ecken in $M(i,j)$ als Kandidaten ausgeschlossen werden.

Induktionsanfang: $(i,j) = (2,1)$. Die Behauptung gilt, weil $M(2,1) = \varnothing$.

Induktionsschluß: Befinde sich der Algorithmus an Postition (i,j), und sei die Menge $M := M(i,j)$ markiert.

1. Fall: $aij = 0$. Wegen Folgerung 2 ist die Ecke j auszuschließen, d. h. es muß gelten $M' = M \cup \{j\}$. Der Algorithmus setzt $i' := i+1$ und $j' := i$. Damit ist

$$\begin{aligned} M' &= M(i',j') = M(i+1,i) = \\ &= \{1, 2, \ldots, i\} \setminus \{i\} = \{1, \ldots, i-1\} = \\ &= \big(\{1, \ldots, i-1\} \setminus \{j\}\big) \cup \{j\} = \\ &= M \cup \{j\} \end{aligned}$$

2. Fall: $a_{ij} = 1$. Wegen Folgerung 2 ist die Ecke i auszuschließen, d. h. es muß gelten $M' = M \cup \{i\}$. Der Algorithmus setzt $i' := i+1$ und $j' := j$.

Damit ist

$$\begin{aligned} M' &= M(i', j') = M(i+1, j) = \\ &= \{1, 2, \dots, i\} \setminus \{j\} = \\ &= \big(\{1, \dots, i-1\} \setminus \{j\}\big) \cup \{i\} = \\ &= M \cup \{i\} \end{aligned}$$

Am Ende des Algorithmus gilt $i = n + 1$, also sind die Ecken

$$M(n+1, j) = \{1, \dots, n\} \setminus \{j\}$$

markiert, und verbleibt nur die Ecke j als Kandidat. $\square$

Offenbar ist die Laufzeit von Algorithmus 3 in $\mathcal{O}(n)$. Nach Ende des Algorithmus ist nur noch eine Ecke j potentieller Kandidat. Ob j dann $g^-(j) = n - 1$ und $g^+(j) = 0$ erfüllt, kann mit Algorithmus 4 in $\mathcal{O}(n)$ Zeit getestet werden.

Algorithmus 4 Testen der (einzigen) Kandidatenecke j.

Kandidat $\leftarrow$ true
for $k \in \{1, 2, \dots, n\} \setminus \{j\}$ **do**
  if $a_{jk} = 1 \vee a_{kj} = 0$ **then**
    Kandidat $\leftarrow$ false
    stop

Aufgabe 2

(a) **Reflexivität** Für alle $v \in V$ gilt nach Definition 3.9: $v \in E_G(v)$, also ist $v \sim v$ nach Definition 3.10.

Symmetrie Sei $u \sim v$. Dann ist $v \in E_G(u)$ und $u \in E_G(v)$. Nach Definition 3.10 ist daher dann auch $v \sim u$.

Transitivität Sei $u \sim v$ und $v \sim x$. Dann ist $v \in E_G(u)$. Entweder ist $u = v$ oder es existiert ein Weg w mit $\alpha(w) = u$ und $\omega(w) = v$.

Ist $v = x$, so ist dann entweder $u = v = x$ oder w ein Weg von u nach x, also $x \in E_G(u)$. Falls $v \neq x$, so muß wegen $x \in E_G(v)$ ein Weg w' mit $\alpha(w') = v$ und $\omega(w')$ existieren. Wenn $v = u$, so ist dieser Weg w' ein Weg von u nach x. Ansonsten ist $w \circ w$ ein Weg, der von u nach x führt. Insgesamt ist damit also $x \in E_G(u)$ gezeigt. Die Umkehrung $u \in E_G(x)$ folgt analog. $\square$

(b) Da G stark zusammenhängend und $|V| \geq 2$ ist, muß für jedes $v \in V$ gelten: $g^+(v) \geq 1$. Somit gilt

$$|R| = \sum_{v \in V} g^+(v) \geq \sum_{v \in V} 1 = |V|.$$

$\square$

Aufgabe 3

Sei $w = (r_1, \ldots, r_k)$ und $s(w) = (u = v_1, \ldots, v_{k+1} = v)$. Falls w bereits elementar ist, so ist nichts zu zeigen. Ansonsten sei $v_i = v_{i+p}$ und i minimal mit der Eigenschaft, daß v_i von w mehr als einmal berührt wird. Dann ist $w' := (r_1, \ldots, r_{i-1}, r_{i+p}, \ldots, r_k)$ ein Weg von u nach v. Falls w' elementar ist, so sind wir fertig. Ansonsten setzen wir das obige Verfahren mit w' fort. Da die Länge der Wege in jedem Schritt jeweils strikt abnimmt, muß das Verfahren mit einem elementaren Weg terminieren. $\square$

Aufgabe 4

Die erste Ungleichung ist trivial. Wir zeigen daher nur den zweiten Teil. Seien $G_1, \ldots, G_p$ die Komponenten von G und $G'_1, \ldots, G'_k$ die Komponenten von $G - e$. Sei ferner $\gamma(e) = \{u, v\}$. Offenbar sind dann u und v in der gleichen Komponenten von G enthalten. Sei dies O.B.d.A. G_1.

Wir zeigen nun, daß für $i = 2, \ldots, p$ folgendes gilt: Sind $a, b \in G_i$, so existiert eine Komponente von $G - e$ die beide Ecken a und b enthält. Danach zeigen wir, daß G_1 durch Entfernen von e in höchstens zwei Komponenten zerfällt. Daraus folgt dann $k \leq p + 1$.

Seien $a, b \in G_i$ für ein $i \geq 2$. Dann existiert nach Definition des Zusammenhangs ein Weg $w = (e_1, \ldots, e_s)$ zwischen a und b in G. Dieser Weg kann e nicht enthalten, denn sonst wären $u, v \in G_i$ im Widerspruch dazu, daß $u, v \in G_1$ und der Tatsache, daß die Komponenten eine Partition von V bilden. Dann ist w aber auch ein Weg in $G - e$ zwischen a und b, d.h. a und b müssen in der gleichen Komponente von $G - e$ enthalten sein.

Wir betrachten nun $G_1 = (V_1, E_1)$, d.h. diejenige Komponente von G, welche u und v enthält. Wir definieren eine Partition $V_1 = U \cup (V_1 \setminus U)$ von V_1 durch

$$U := \{u\} \cup \{\, x \in V_1 : \text{es gibt einen Weg von } x \text{ nach } u, \text{ der } e \text{ nicht benutzt} \,\}.$$

Offenbar sind dann alle $x \in U$ in der gleichen Zusammenhangskomponente von $G - e$ wie u. Wir zeigen, daß alle Ecken aus $V_1 \setminus U$ in der gleichen Zusammenhangskomponente wie v sind.

Sei dazu $x \in V_1 \setminus U$ beliebig. Nach Voraussetzung existiert ein Weg $w = (e_1, \ldots, e_s)$ von x nach u in G. Dieser Weg benutzt e. O.B.d.A. können wir annehmen, daß $e = e_s$ und e sonst nicht in w vorkommt. Dann ist $(e_1, \ldots, e_{s-1})$ ein Weg von x nach v in $G - e$.