

GBI Tutorium Nr. 2⁵

Tutorium 8

Dominik Muth - dominik.muth@student.kit.edu | 12. Dezember 2012

INSTITUT FÜR INFORMATIK



- 1 Wiederholung
- 2 Graphen im Rechner
 - Adjazenzmatrix
 - Adjazenzliste
 - Wegematrix
 - 2-Erreichbarkeitsrelation
 - Algorithmus von Warshall
- 3 Fragen

Hinweis

- Habt ihr euch für den Übungsschein angemeldet?
- Und für die Klausur?

Hinweis

- Habt ihr euch für den Übungsschein angemeldet?
- Und für die Klausur?

- 1 Wiederholung
- 2 Graphen im Rechner
- 3 Fragen

- Zwei unterschiedliche Graphen können das gleiche Aussagen.
- Ein Teilgraph ist definiert mit $V' \subseteq V, E' \subseteq V' \times V'$
- In jedem gerichteten Baum gibt es genau einen Knoten x_0 mit $d^+(x_0) = 0 \wedge d^-(x_0) \geq 0$
- Zwischen zwei isomorphen Graphen gibt es immer nur einen Isomorphismus

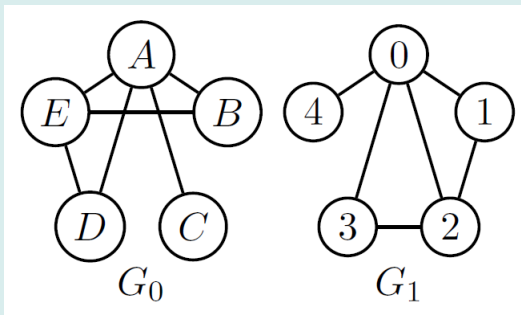
- Zwei unterschiedliche Graphen können das gleiche Aussagen. ✓
- Ein Teilgraph ist definiert mit $V' \subseteq V, E' \subseteq V' \times V'$
- In jedem gerichteten Baum gibt es genau einen Knoten x_0 mit $d^+(x_0) = 0 \wedge d^-(x_0) \geq 0$
- Zwischen zwei isomorphen Graphen gibt es immer nur einen Isomorphismus

- Zwei unterschiedliche Graphen können das gleiche Aussagen. ✓
- Ein Teilgraph ist definiert mit $V' \subseteq V, E' \subseteq V' \times V'$ ✗
- In jedem gerichteten Baum gibt es genau einen Knoten x_0 mit $d^+(x_0) = 0 \wedge d^-(x_0) \geq 0$
- Zwischen zwei isomorphen Graphen gibt es immer nur einen Isomorphismus

- Zwei unterschiedliche Graphen können das gleiche Aussagen. ✓
- Ein Teilgraph ist definiert mit $V' \subseteq V, E' \subseteq V' \times V'$ ✗
- In jedem gerichteten Baum gibt es genau einen Knoten x_0 mit $d^+(x_0) = 0 \wedge d^-(x_0) \geq 0$ ✓
- Zwischen zwei isomorphen Graphen gibt es immer nur einen Isomorphismus ✗

Isomorphie

Geben Sie alle Isomorphismen zwischen G_0 und G_1 an.



Graphen: ÜB7 (WS08/09)

Gegeben sei der Graph $G = (V, E)$ mit $V = \{0, 1\}^3$ und $E = \{(xw, wy) \mid x, y \in \{0, 1\} \wedge w \in \{0, 1\}^2\}$.

- Zeichnen Sie den Graphen
- Geben Sie einen Zyklus in G an, der außer dem Anfangs- und Endknoten jeden Knoten von G genau einmal enthält.
- Geben Sie einen geschlossenen Pfad in G an, der jede Kante von G genau einmal enthält.

1 Wiederholung

2 Graphen im Rechner

- Adjazenzmatrix
- Adjazenzliste
- Wegematrix
- 2-Erreichbarkeitsrelation
- Algorithmus von Warshall

3 Fragen

- Wie werden Graphen gespeichert?
- Wie effizient sind die Methoden?
- Was verstehen wir unter Effizienz?

Was ist das?

Ein Graph kann in einer $n \times n$ Matrix gespeichert werden, wobei gilt:
 $n = |V|$.

Wie kann man sich dies Vorstellen?

Für jede Kante (x, y) existiert eine 1 in der Matrix an der Stelle (x, y) .

Was ist das?

Ein Graph kann in einer $n \times n$ Matrix gespeichert werden, wobei gilt:
 $n = |V|$.

Wie kann man sich dies Vorstellen?

Für jede Kante (x, y) existiert eine 1 in der Matrix an der stelle (x, y) .

Was ist das?

Ein Graph kann in einer $n \times n$ Matrix gespeichert werden, wobei gilt:
 $n = |V|$.

Wie kann man sich dies Vorstellen?

Für jede Kante (x, y) existiert eine 1 in der Matrix an der stelle (x, y) .

Was ist das?

Ein Graph kann in einer $n \times n$ Matrix gespeichert werden, wobei gilt:
 $n = |V|$.

Wie kann man sich dies Vorstellen?

Für jede Kante (x, y) existiert eine 1 in der Matrix an der Stelle (x, y) .

Beispiel

Wie sieht die Adjazenzmatrix für folgenden gerichteten Graphen aus:

$$G = (V, E)$$

$$V = \{0, 1, 2, 3, 4\}$$

$$E = \{(0, 0), (0, 1), (0, 2), (0, 4), (2, 3), (4, 1), (2, 0)\}$$

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Einige Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Woran erkennt man Schlingen?
- Welche Eigenschaft haben Adjazenzmatrizen ungerichteter Graphen?

Gegeben sei folgende Adjazenzmatrix:

$$A = \begin{array}{c} \begin{array}{cccc} & 0 & 1 & 2 & 3 \\ \begin{array}{c} 0 \\ 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \end{array}$$

Zeichnen Sie den Graphen zur Adjazenzmatrix A.

Was ist das?

Bei einer Adjazenzliste wird für jeden Knoten einzeln gespeichert, welche ausgehenden Kanten dieser besitzt.

Beispiel

Wie sieht die Adjazenzliste für den Graphen von vorhin aus:

$$G = (V, E)$$

$$V = \{0, 1, 2, 3, 4\}$$

$$E = \{(0, 0), (0, 1), (0, 2), (0, 4), (2, 3), (4, 1), (2, 0)\}$$

Nochmal Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Sind Adjazenzlisten besser als Adjazenzmatrizen?

Nochmal Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Sind Adjazenzlisten besser als Adjazenzmatrizen?

Nochmal Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Sind Adjazenzlisten besser als Adjazenzmatrizen?

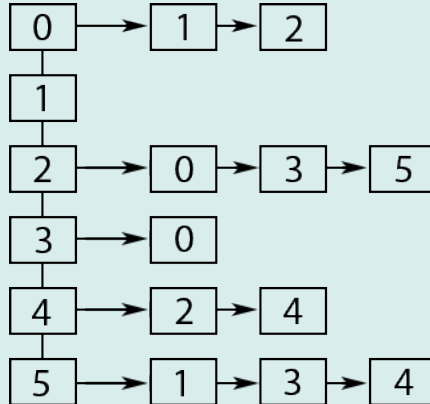
Nochmal Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Sind Adjazenzlisten besser als Adjazenzmatrizen?

Nochmal Denkanstöße:

- Wie sieht es mit dem Speicherplatz aus?
- Wie "schnell" kann man herausfinden, ob eine Kante zwischen x und y existiert?
- Wie schnell hat man Zugriff auf alle adjazenten Knoten von einem Knoten x ?
- Sind Adjazenzlisten besser als Adjazenzmatrizen?

Zeichnen sie den Graphen zur folgenden Adjazenzliste:



Wozu?

Die Wegematrix sagt uns, ob ein Pfad zwischen zwei Knoten existiert.

Beispiel

Zeichnen Sie die Wegematrix zum Graphen, der an der Tafel stehen sollte.

- ◀ ◻ ▶ ◀ ◼ ▶ ◀ ≡ ▶ ◀ ≡ ▶ ≡

- $$W = A$$

- $$W = A$$

- Wie sieht die Wegematrix aus, wenn $A = \text{alles Eins}$?
 $W = A$
- Wann ist allgemein $W = A$?
Wenn Kantenrelationen reflexiv und transitiv

- Wie sieht die Wegematrix aus, wenn $A = \text{alles Eins}$?
 $W = A$
- Wann ist allgemein $W = A$?
Wenn Kantenrelationen reflexiv und transitiv
- Warum nicht symmetrisch?

- Wie sieht die Wegematrix aus, wenn $A = \text{alles Eins}$?
 $W = A$
- Wann ist allgemein $W = A$?
Wenn Kantenrelationen reflexiv und transitiv
- Warum nicht symmetrisch?
Auch Wegematrizen können asymmetrisch sein.

Was ist das?

Über die 2-Erreichbarkeitsrelation können wir herausfinden, welche Knoten über 2 Kanten erreichbar sind.

Wie gehts das?

Indem wir die Adjazenzmatrix mit sich selbst mal nehmen. (A^2)

Was ist das?

Über die 2-Erreichbarkeitsrelation können wir herausfinden, welche Knoten über 2 Kanten erreichbar sind.

Wie gehts das?

Indem wir die Adjazenzmatrix mit sich selbst mal nehmen. (A^2)

Definition

Wir definieren x_{ij} als das Element der Matrix X , welches in der i -ten Zeile und j -ten Spalte.

Dann lässt sich die Multiplikation der Matrizen A und B folgendermaßen darstellen:

$$\forall c \in C \mid c_{ij} = \sum_{k=1}^m a_{ik} \cdot b_{kj}$$

mit $b \in B$, $a \in A$ und m sei die Breite der Matrix A und die Höhe der Matrix B .

Beispiel

$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \cdot \begin{pmatrix} 6 & -1 \\ 3 & 2 \\ 0 & -3 \end{pmatrix} =$$

$$\begin{pmatrix} 1 \cdot 6 + 2 \cdot 3 + 3 \cdot 0 & 1 \cdot (-1) + 2 \cdot 2 + 3 \cdot (-3) \\ 4 \cdot 6 + 5 \cdot 3 + 6 \cdot 0 & 4 \cdot (-1) + 5 \cdot 2 + 6 \cdot (-3) \end{pmatrix} = \begin{pmatrix} 12 & -6 \\ 39 & -12 \end{pmatrix}$$

Was ist das?

Über die 2-Erreichbarkeitsrelation können wir herausfinden, welche Knoten über 2 Kanten erreichbar sind.

Wie gehts das?

Indem wir die Adjazenzmatrix mit sich selbst mal nehmen. (A^2)

Warum funktioniert das?

siehe Beispiel an der Tafel

Wie können wir die 2-Erreichbarkeitsrelation also nutzen, um die Wegematrix aufzustellen?

Geht es auch einfacher/schneller?

Wie können wir die 2-Erreichbarkeitsrelation also nutzen, um die Wegematrix aufzustellen?

Geht es auch einfacher/schneller?

Definition

Der Warshall-Algorithmus ist ein Algorithmus zum Berechnen von Wegematrizen.

Algorithmus Teil 1

```
for  $i \leftarrow 0$  to  $n - 1$  do  
  for  $j \leftarrow 0$  to  $n - 1$  do  
    if  $i = j$  then  
       $W[i, j] \leftarrow 1$ ;  
    else  
       $W[i, j] \leftarrow A[i, j]$   
    end if  
  od  
od
```

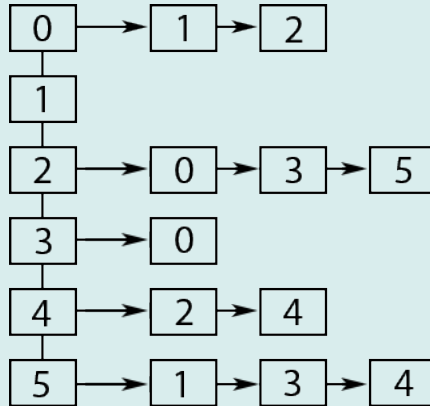
Algorithmus Teil 2

```
for  $k \leftarrow 0$  to  $n - 1$  do
  for  $i \leftarrow 0$  to  $n - 1$  do
    for  $j \leftarrow 0$  to  $n - 1$  do
       $W[i, j] \leftarrow \max(W[i, j]; \min(W[i, k]; W[k, j]))$ 
    od
  od
od
```

Beispiel

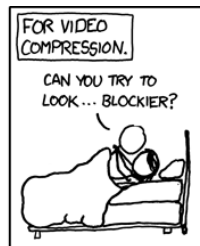
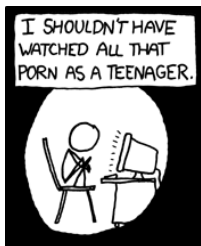
$G = (\{0, 1, 2, 3\}, \{(0, 3), (1, 0), (2, 3), (3, 1)\})$

Gegeben sei eine Adjazenzliste, bilden Sie die Adjazenzmatrix und bestimmen Sie mit Hilfe des Warshall-Algorithmus die Wegematrix.



- 1 Wiederholung
- 2 Graphen im Rechner
- 3 Fragen**

- Fragen zum Stoff?
- Fragen zum nächsten Übungsblatt?
- Generelle Fragen?
- Feedback?



source : <http://imgs.xkcd.com/comics/porn.png>