Graphen

Graphen

Bei Modellierungsaufgaben geht es oft darum,

Objekte *sowie* ***Beziehungen zwischen je zwei Objekten*** zu

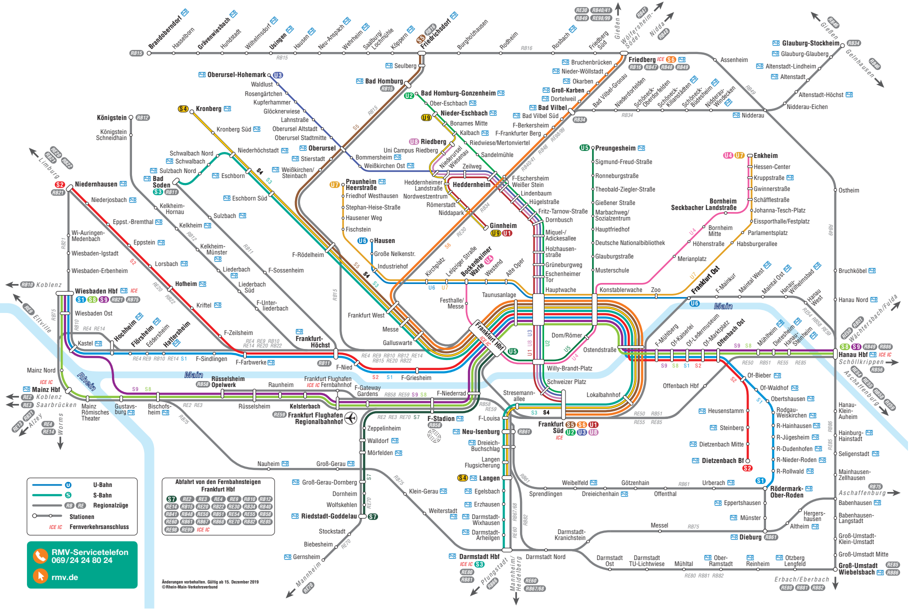
beschreiben: Graphen sind dafür maßgeschneidert.

Ein Graph besteht aus

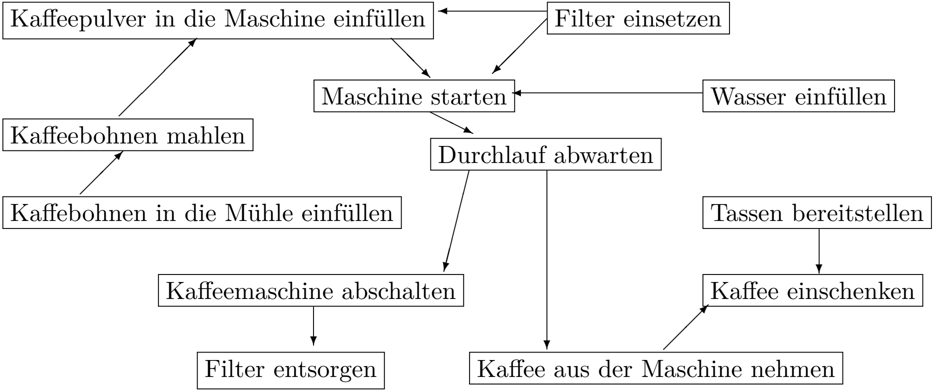
- „**Knoten**“ (repräsentieren Objekte) und
- „**Kanten**“ (repräsentieren Beziehungen zwischen je zwei Objekten).

Ein erstes Beispiel: Der Netzplan der Frankfurter S- und U-Bahnen zeigt U- und S-Bahn Stationen (als Knoten) und Direktverbindungen zwischen den Stationen (als Kanten).

Schnellbahnplan



Kaffeekochen



Ungerichtete Graphen

Was ist ein ungerichteter Graph $G = (V, E)$?

DEFINITION

Ein **ungerichteter Graph** $G = (V, E)$ besteht aus der Knotenmenge V und der Kantenmenge E mit

$$E \quad \subseteq \quad \big\{\{i,j\} \,:\, i \in V, j \in V, i \neq j\big\}.$$

Die Elemente aus V heißen **Knoten** von G, die Elemente aus E heißen **(ungerichtete) Kanten** von G:

Kanten sind also **2-elementige Teilmengen** von V.

Was ist ein ungerichteter Graph $G = (V, E)$?

DEFINITION

Ein **ungerichteter Graph** $G = (V, E)$ besteht aus der Knotenmenge V und der Kantenmenge E mit

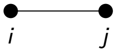
$$E \subseteq \Big\{\{i,j\} : i \in V, j \in V, i \neq j\Big\}.$$

Die Elemente aus V heißen **Knoten** von G, die Elemente aus E heißen **(ungerichtete) Kanten** von G:

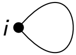
Kanten sind also **2-elementige Teilmengen** von V.

Es gibt zwischen zwei Knoten i und j aus V

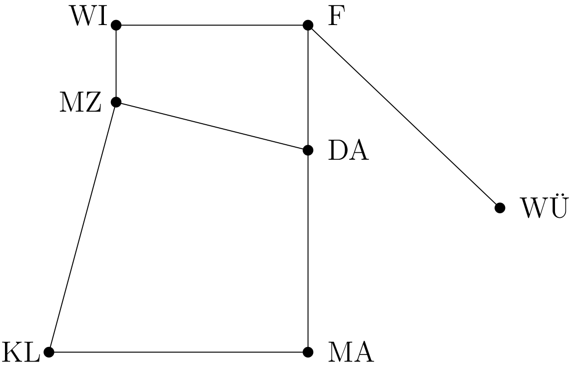
- **höchstens** eine Kante $\{i, j\}$, die grafisch dargestellt wird durch



- **keine** Kante, falls $i = j$ ist. Nicht erlaubt sind somit „Schleifen".



Autobahnverbindungen

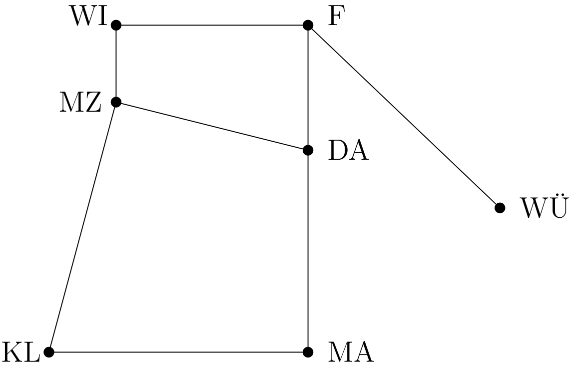


DA	$\widehat{=}$	Darmstadt
F	$\widehat{=}$	Frankfurt
KL	$\widehat{=}$	Kaiserslautern
MA	$\widehat{=}$	Mannheim
MZ	$\widehat{=}$	Mainz
WI	$\widehat{=}$	Wiesbaden
WÜ	$\widehat{=}$	Würzburg

$G = (V, E)$ ist ein ungerichteter Graph mit

$$V \quad :=$$

Autobahnverbindungen



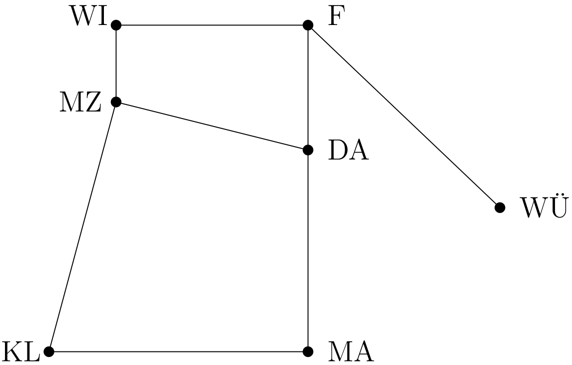
DA	$\widehat{=}$	Darmstadt
F	$\widehat{=}$	Frankfurt
KL	$\widehat{=}$	Kaiserslautern
MA	$\widehat{=}$	Mannheim
MZ	$\widehat{=}$	Mainz
WI	$\widehat{=}$	Wiesbaden
WÜ	$\widehat{=}$	Würzburg

$G = (V, E)$ ist ein ungerichteter Graph mit

$$V := \{\text{MZ}, \text{WI}, \text{MA}, \text{DA}, \text{KL}, \text{F}, \text{WÜ}\} \text{ und}$$

$$E :=$$

Autobahnverbindungen



DA	$\widehat{=}$	Darmstadt
F	$\widehat{=}$	Frankfurt
KL	$\widehat{=}$	Kaiserslautern
MA	$\widehat{=}$	Mannheim
MZ	$\widehat{=}$	Mainz
WI	$\widehat{=}$	Wiesbaden
WÜ	$\widehat{=}$	Würzburg

$G = (V, E)$ ist ein ungerichteter Graph mit

$$\begin{aligned} V &:= \{\text{MZ}, \text{WI}, \text{MA}, \text{DA}, \text{KL}, \text{F}, \text{WÜ}\} \quad \text{und} \\ E &:= \{\{\text{MZ}, \text{WI}\}, \{\text{WI}, \text{F}\}, \{\text{F}, \text{DA}\}, \{\text{F}, \text{WÜ}\}, \{\text{MZ}, \text{DA}\}, \\ &\quad\; \{\text{MZ}, \text{KL}\}, \{\text{KL}, \text{MA}\}, \{\text{DA}, \text{MA}\}\}. \end{aligned}$$

(Eine Kante $\{u, v\}$ gehört zu E, wenn die u, v entsprechenden Städte direkt verbunden sind.)

Ungerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph.

- Ein Knoten $v \in V$ heißt **inzident** mit einer Kante $e \in E$, falls $v \in e$.

Ungerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph.

- Ein Knoten $v \in V$ heißt **inzident** mit einer Kante $e \in E$, falls $v \in e$.
- Die beiden mit einer Kante $e \in E$ inzidenten Knoten nennen wir die **Endknoten** von e, und sagen, dass e diese beiden Knoten **verbindet**.
  - Zwei Knoten $v, v' \in V$ heißen **benachbart** (bzw. **adjazent**), falls es eine Kante $e \in E$ mit Endknoten v und v' gibt (d.h. $e = \{v, v'\}$).
  - Falls v und v' zwei benachbarte Knoten sind, so sagen wir auch, dass v' ein **Nachbar** von Knoten v ist.

Ungerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph.

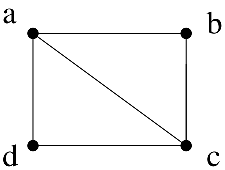
- Ein Knoten $v \in V$ heißt **inzident** mit einer Kante $e \in E$, falls $v \in e$.
- Die beiden mit einer Kante $e \in E$ inzidenten Knoten nennen wir die **Endknoten** von e, und sagen, dass e diese beiden Knoten **verbindet**.
  - Zwei Knoten $v, v' \in V$ heißen **benachbart** (bzw. **adjazent**), falls es eine Kante $e \in E$ mit Endknoten v und v' gibt (d.h. $e = \{v, v'\}$).
  - Falls v und v' zwei benachbarte Knoten sind, so sagen wir auch, dass v' ein **Nachbar** von Knoten v ist.
- Der **Grad von v in G** (kurz: $\mathrm{Grad}_G(v)$), ist die Anzahl der Kanten, die v als Endknoten haben, d.h.

$$\mathrm{Grad}_G(v) \;=\; |\{e \in E \;:\; v \in e\}|.$$

  Der **Grad** von G (kurz: $\mathrm{Grad}(G)$) ist der **maximale** Grad eines Knotens in G.

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$

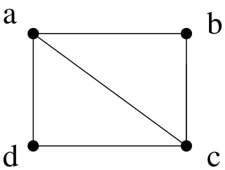


gilt:

$$\mathrm{Grad}_G(a) =$$

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$

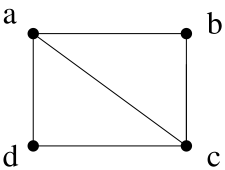


gilt:

$$\begin{aligned}\mathrm{Grad}_G(a) &= 3\\ \mathrm{Grad}_G(b) &= \end{aligned}$$

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$

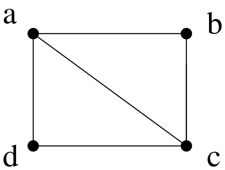


gilt:

$$\begin{aligned} \mathrm{Grad}_G(a) &= 3 \\ \mathrm{Grad}_G(b) &= 2 \\ \mathrm{Grad}_G(c) &= \end{aligned}$$

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$

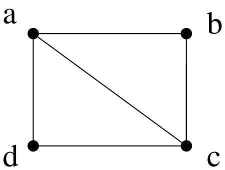


gilt:

$$\begin{aligned}
\text{Grad}_G(a) &= 3\\
\text{Grad}_G(b) &= 2\\
\text{Grad}_G(c) &= 3\\
\text{Grad}_G(d) &=
\end{aligned}$$

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$

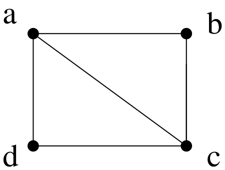


gilt:

$$\begin{aligned} \text{Grad}_G(a) &= 3 \\ \text{Grad}_G(b) &= 2 \\ \text{Grad}_G(c) &= 3 \\ \text{Grad}_G(d) &= 2 \end{aligned} \quad \text{und} \quad \text{Grad}(G) =$$

Der Knotengrad: Ein Beispiel

Für den Graphen $G =$



gilt:

$$\begin{array}{l} \mathrm{Grad}_G(a) = 3 \\ \mathrm{Grad}_G(b) = 2 \\ \mathrm{Grad}_G(c) = 3 \\ \mathrm{Grad}_G(d) = 2 \end{array} \quad \text{und} \quad \mathrm{Grad}(G) = 3.$$

Python: Eine Klasse „Graph“ (1/2)

Repräsentiere $G = (V, E)$ durch ein

Dictionary „Graph“,

das für jeden Knoten $v \in V$ eine Liste Graph$[v]$ der Nachbarn von v besitzt.

Für den Graphen aus der letzten Folie ist

$$\begin{aligned} \text{Graph} = \{ \quad & \text{“}a\text{“} && : [\text{“}d\text{“}, \text{“}b\text{“}, \text{“}c\text{“}], \\ & \text{“}b\text{“} && : [\text{“}a\text{“}, \text{“}c\text{“}], \\ & \text{“}c\text{“} && : [\text{“}d\text{“}, \text{“}a\text{“}, \text{“}b\text{“}], \\ & \text{“}d\text{“} && : [\text{“}c\text{“}, \text{“}a\text{“}] \\ \} & \end{aligned}$$

Python: Eine Klasse „Graph“ (2/2)

Mögliche Operationen der Klasse:

1. Initialisiere den Graphen.
2. Füge Knoten (Kanten) hinzu, bzw. entferne Knoten (Kanten).
3. Bestimme alle von einem Knoten $v \in V$ aus *erreichbaren* Knoten bzw. bestimme *kürzeste Wege* von v zu allen erreichbaren Knoten.
4. Bestimme die *Zusammenhangskomponenten* des Graphen.
5. Wenn Kanten (unterschiedlich) „teuer“ sind:
   Bestimme eine billigste „aufspannende“ Teilmenge $F \subseteq E$ von Kanten, so dass alle auf E erreichbaren Knotenpaare auch auf F erreichbar bleiben.
6. ...?!

Erreichbarkeit, kürzeste Wege, Zusammenhangskomponenten: Was ist das?

Gerichtete Graphen

Was ist ein gerichteter Graph $G = (V, E)$?

DEFINITION

Ein **gerichteter Graph** $G = (V, E)$ besteht aus der Knotenmenge V und der Kantenmenge E mit

$$E \subseteq \{(i,j) \ : \ i \in V, j \in V\}.$$

Die Elemente aus V heißen **Knoten**, die Elemente aus E heißen **(gerichtete) Kanten** von G.

Kanten sind also **geordnete** Paare von Knoten, d.h. Elemente von $V \times V$.

Was ist ein gerichteter Graph $G = (V, E)$?

DEFINITION

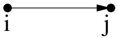
Ein **gerichteter Graph** $G = (V, E)$ besteht aus der Knotenmenge V und der Kantenmenge E mit

$$E \subseteq \{(i, j) : i \in V, j \in V\}.$$

Die Elemente aus V heißen **Knoten**, die Elemente aus E heißen **(gerichtete) Kanten** von G.

Kanten sind also **geordnete** Paare von Knoten, d.h. Elemente von $V \times V$.

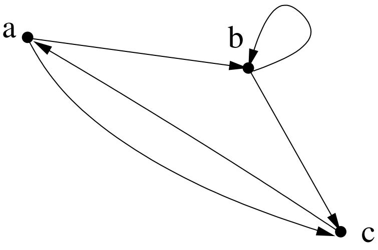
In der grafischen Darstellung eines Graphen stellen wir die Kante (i, j) als Pfeil **von** Knoten i **nach** Knoten j dar, also



Beachte, dass wir „Schleifen“ (i, i) diesmal zulassen.

Gerichtete Graphen: ein erstes Beispiel

Der gerichtete Graph $G = (V, E)$ mit

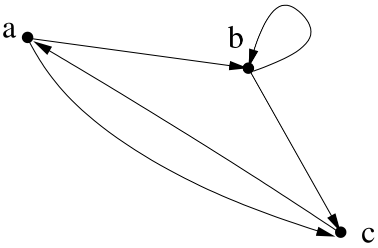


wird repräsentiert durch

$$V \quad :=$$

Gerichtete Graphen: ein erstes Beispiel

Der gerichtete Graph $G = (V, E)$ mit

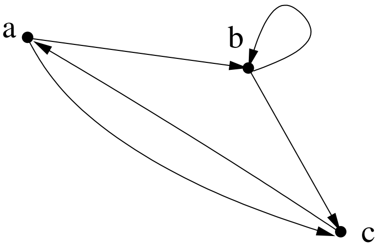


wird repräsentiert durch

$$\begin{aligned} V &:= \{a, b, c\} \text{ und} \\ E &:= \end{aligned}$$

Gerichtete Graphen: ein erstes Beispiel

Der gerichtete Graph $G = (V, E)$ mit



wird repräsentiert durch

$$\begin{array}{rcl} V & := & \{a, b, c\} \text{ und} \\ E & := & \{(a, b), (b, b), (b, c), (c, a), (a, c)\}. \end{array}$$

Darstellungen von Graphen

- **Abstrakt**, durch Angabe der Knotenmenge V und der Kantenmenge E.
  Beispiel: $G_1 = (V_1, E_1)$ mit $V_1 := \{a, b, c, d\}$ und

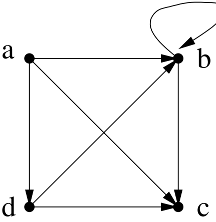
$$E_1 \quad := \quad \{(a,b), (a,c), (a,d), (b,b), (b,c), (d,b), (d,c)\}.$$

Darstellungen von Graphen

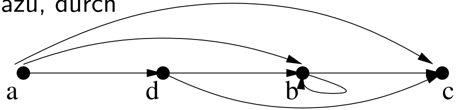
- **Abstrakt**, durch Angabe der Knotenmenge V und der Kantenmenge E.
  Beispiel: $G_1 = (V_1, E_1)$ mit $V_1 := \{a, b, c, d\}$ und

$$E_1 \quad := \quad \{(a,b), (a,c), (a,d), (b,b), (b,c), (d,b), (d,c)\}.$$

- Der obige Graph G_1 kann **grafisch** dargestellt werden durch



oder, äquivalent dazu, durch

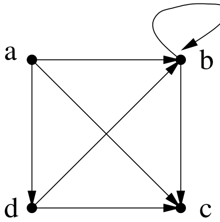


Darstellungen von Graphen

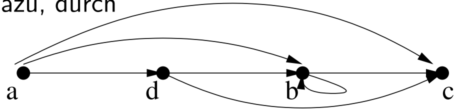
- **Abstrakt**, durch Angabe der Knotenmenge V und der Kantenmenge E.
  Beispiel: $G_1 = (V_1, E_1)$ mit $V_1 := \{a, b, c, d\}$ und

$$E_1 \quad := \quad \{(a,b),(a,c),(a,d),(b,b),(b,c),(d,b),(d,c)\}.$$

- Der obige Graph G_1 kann **grafisch** dargestellt werden durch



  oder, äquivalent dazu, durch



- Durch **Adjazenzlisten** und **Adjazenzmatrizen**: Siehe die Vorlesung „**Algorithmen und Datenstrukturen 1**" im zweiten Semester.

Gerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein gerichteter Graph.

- Ist $e = (i, j) \in E$, so ist i der **Ausgangsknoten** von e und j der **Endknoten** von e, und wir sagen, dass i und j **benachbart** sind.
  - i heißt ein **direkter Vorgänger** von j und j ein **direkter Nachfolger** von i.

  Eine Kante der Form (i, i) wird **Schleife** genannt. D.h.: Eine Schleife ist eine Kante, deren Ausgangs- und Endknoten identisch ist.

Gerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein gerichteter Graph.

- Ist $e = (i, j) \in E$, so ist i der **Ausgangsknoten** von e und j der **Endknoten** von e, und wir sagen, dass i und j **benachbart** sind.
  - i heißt ein **direkter Vorgänger** von j und j ein **direkter Nachfolger** von i.

  Eine Kante der Form (i, i) wird **Schleife** genannt. D.h.: Eine Schleife ist eine Kante, deren Ausgangs- und Endknoten identisch ist.
- Ein Knoten $v \in V$ heißt **inzident** mit einer Kante $e \in E$, falls v der Ausgangs- oder der Endknoten von e ist.

Gerichtete Graphen: Wichtige Begriffe

DEFINITION

Sei $G = (V, E)$ ein gerichteter Graph.

- Ist $e = (i, j) \in E$, so ist i der **Ausgangsknoten** von e und j der **Endknoten** von e, und wir sagen, dass i und j **benachbart** sind.
  - i heißt ein **direkter Vorgänger** von j und j ein **direkter Nachfolger** von i.

  Eine Kante der Form (i, i) wird **Schleife** genannt. D.h.: Eine Schleife ist eine Kante, deren Ausgangs- und Endknoten identisch ist.
- Ein Knoten $v \in V$ heißt **inzident** mit einer Kante $e \in E$, falls v der Ausgangs- oder der Endknoten von e ist.
- Der **Ausgangsgrad** von v in G (kurz: $\text{Aus-Grad}_G(v)$) ist die Anzahl der Kanten mit v als Ausgangsknoten. D.h.:

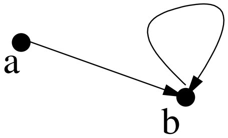
  $$\text{Aus-Grad}_G(v) \; = \; |\{e \in E \; : \; \text{es ex. } v' \in V \text{ s.d. } e = (v, v')\}|.$$

  Der **Eingangsgrad** von v in G (kurz $\text{Ein-Grad}_G(v)$) ist die Anzahl der Kanten mit v als Eingangsknoten. D.h.:

  $$\text{Ein-Grad}_G(v) \; = \; |\{e \in E \; : \; \text{es ex. } v' \in V \text{ s.d. } e = (v', v)\}|.$$

Ein Beispiel

Für den Graphen $G =$

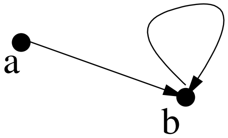


gilt:

$$\text{Ein-Grad}_G(a) \quad =$$

Ein Beispiel

Für den Graphen G =

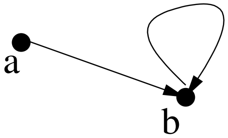


gilt:

$$\begin{aligned} \text{Ein-Grad}_G(a) &= 0 \\ \text{Ein-Grad}_G(b) &= \end{aligned}$$

Ein Beispiel

Für den Graphen $G =$

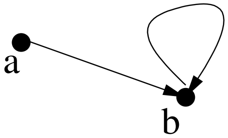


gilt:

$$
\begin{aligned}
\text{Ein-Grad}_G(a) &= 0 \\
\text{Ein-Grad}_G(b) &= 2 \\
\text{Aus-Grad}_G(a) &=
\end{aligned}
$$

Ein Beispiel

Für den Graphen $G =$

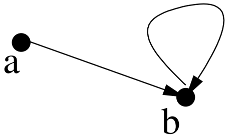


gilt:

$$\begin{aligned} \text{Ein-Grad}_G(a) &= 0 \\ \text{Ein-Grad}_G(b) &= 2 \\ \text{Aus-Grad}_G(a) &= 1 \\ \text{Aus-Grad}_G(b) &= \end{aligned}$$

Ein Beispiel

Für den Graphen $G =$



gilt:

$$\begin{aligned} \text{Ein-Grad}_G(a) &= 0 \\ \text{Ein-Grad}_G(b) &= 2 \\ \text{Aus-Grad}_G(a) &= 1 \\ \text{Aus-Grad}_G(b) &= 1. \end{aligned}$$

Wege in Graphen

Wege

DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ungerichteter) Graph.

Ein **Weg** in G der **Länge** ℓ ist ein Tupel

$$(v_0, \ldots, v_\ell) \ \in \ V^{\ell+1},$$

für ein $\ell \in \mathbb{N}$, so dass für alle $i \in \mathbb{N}$ mit $0 \leqslant i < \ell$ gilt:

- falls G ein gerichteter Graph ist, so ist

Wege

DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ungerichteter) Graph.

Ein **Weg** in G der **Länge** ℓ ist ein Tupel

$$(v_0, \ldots, v_\ell) \in V^{\ell+1},$$

für ein $\ell \in \mathbb{N}$, so dass für alle $i \in \mathbb{N}$ mit $0 \leqslant i < \ell$ gilt:

- falls G ein gerichteter Graph ist, so ist $(v_i, v_{i+1}) \in E$,
- falls G ein ungerichteter Graph ist, so ist $\{v_i, v_{i+1}\} \in E$.

Das Tupel $(v_0, \ldots, v_\ell)$ wird dann ein Weg **der Länge** ℓ **von** v_0 **nach** v_ℓ genannt.

Wege

DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ungerichteter) Graph.

Ein **Weg** in G der **Länge** ℓ ist ein Tupel

$$(v_0, \dots, v_\ell) \ \in \ V^{\ell+1},$$

für ein $\ell \in \mathbb{N}$, so dass für alle $i \in \mathbb{N}$ mit $0 \leqslant i < \ell$ gilt:

- falls G ein gerichteter Graph ist, so ist $(v_i, v_{i+1}) \in E$,
- falls G ein ungerichteter Graph ist, so ist $\{v_i, v_{i+1}\} \in E$.

Das Tupel $(v_0, \dots, v_\ell)$ wird dann ein Weg **der Länge** ℓ **von** v_0 **nach** v_ℓ genannt.

Die Länge des Weges gibt **nicht** an, **wie viele Knoten**, sondern

wie viele Kanten

auf dem Weg durchlaufen werden.

(Einfache) Wege und Kreise

DEFINITION

- Ein Weg $(v_0, \ldots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \ldots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \ldots, v_\ell\}| = \ell + 1$ gilt).

(Einfache) Wege und Kreise

DEFINITION

- Ein Weg $(v_0, \dots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \dots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \dots, v_\ell\}| = \ell + 1$ gilt).
- Ein Weg $(v_0, \dots, v_\ell)$ heißt **Kreis**, wenn $\ell \geqslant 1$ und $v_\ell = v_0$ ist.

(Einfache) Wege und Kreise

DEFINITION

- Ein Weg $(v_0, \ldots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \ldots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \ldots, v_\ell\}| = \ell + 1$ gilt).
- Ein Weg $(v_0, \ldots, v_\ell)$ heißt **Kreis**, wenn $\ell \geqslant 1$ und $v_\ell = v_0$ ist.
- Ein Kreis $(v_0, \ldots, v_\ell)$ heißt **einfacher Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein einfacher Weg ist und *keine Kante mehrfach durchlaufen wird*.

In einem **gerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \ldots, v_\ell)$, für die gilt: $\ell \geqslant$

(Einfache) Wege und Kreise

DEFINITION

- Ein Weg $(v_0, \dots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \dots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \dots, v_\ell\}| = \ell + 1$ gilt).
- Ein Weg $(v_0, \dots, v_\ell)$ heißt **Kreis**, wenn $\ell \geqslant 1$ und $v_\ell = v_0$ ist.
- Ein Kreis $(v_0, \dots, v_\ell)$ heißt **einfacher Kreis**, wenn $(v_0, \dots, v_{\ell-1})$ ein einfacher Weg ist und *keine Kante mehrfach durchlaufen wird*.

In einem **gerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \dots, v_\ell)$, für die gilt: $\ell \geqslant 1$ und $v_\ell = v_0$ und $|\{v_0, \dots, v_{\ell-1}\}| = \ell$.

(Einfache) Wege und Kreise

DEFINITION

- Ein Weg $(v_0, \ldots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \ldots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \ldots, v_\ell\}| = \ell + 1$ gilt).
- Ein Weg $(v_0, \ldots, v_\ell)$ heißt **Kreis**, wenn $\ell \geqslant 1$ und $v_\ell = v_0$ ist.
- Ein Kreis $(v_0, \ldots, v_\ell)$ heißt **einfacher Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein einfacher Weg ist und *keine Kante mehrfach durchlaufen wird*.

In einem **gerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \ldots, v_\ell)$, für die gilt: $\ell \geqslant 1$ und $v_\ell = v_0$ und $|\{v_0, \ldots, v_{\ell-1}\}| = \ell$.

In einem **ungerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \ldots, v_\ell)$, für die gilt: $\ell \geqslant$

(Einfache) Wege und Kreise

DEFINITION

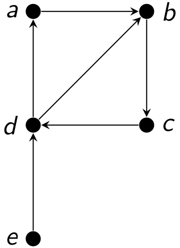
- Ein Weg $(v_0, \dots, v_\ell)$ heißt **einfacher Weg**, wenn kein Knoten mehr als einmal in dem Weg vorkommt (d.h. die Knoten $v_0, \dots, v_\ell$ sind paarweise verschieden, bzw. $|\{v_0, \dots, v_\ell\}| = \ell + 1$ gilt).
- Ein Weg $(v_0, \dots, v_\ell)$ heißt **Kreis**, wenn $\ell \geqslant 1$ und $v_\ell = v_0$ ist.
- Ein Kreis $(v_0, \dots, v_\ell)$ heißt **einfacher Kreis**, wenn $(v_0, \dots, v_{\ell-1})$ ein einfacher Weg ist und *keine Kante mehrfach durchlaufen wird*.

In einem **gerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \dots, v_\ell)$, für die gilt: $\ell \geqslant 1$ und $v_\ell = v_0$ und $|\{v_0, \dots, v_{\ell-1}\}| = \ell$.

In einem **ungerichteten** Graphen G sind **einfache** Kreise genau die Wege der Form $(v_0, \dots, v_\ell)$, für die gilt: $\ell \geqslant 3$ und $v_\ell = v_0$ und $|\{v_0, \dots, v_{\ell-1}\}| = \ell$.

Wege und Kreise: Beispiele (1/2)

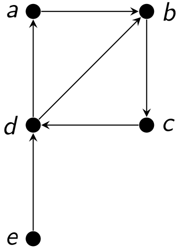
Für den Graphen



gilt:

- (e, d, b, c, d) ist

Für den Graphen

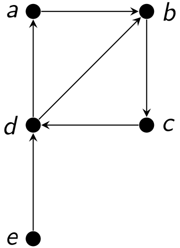


gilt:

- (e, d, b, c, d) ist ein Weg der Länge 4, aber kein einfacher Weg.
- (d, b, c, d) ist

Wege und Kreise: Beispiele (1/2)

Für den Graphen

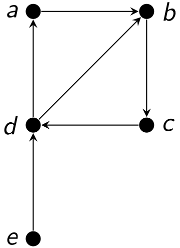


gilt:

- (e, d, b, c, d) ist ein Weg der Länge 4, aber kein einfacher Weg.
- (d, b, c, d) ist ein einfacher Kreis.
- (e, d, a, b) ist

Wege und Kreise: Beispiele (1/2)

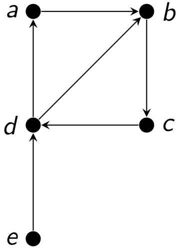
Für den Graphen



gilt:

- (e, d, b, c, d) ist ein Weg der Länge 4, aber kein einfacher Weg.
- (d, b, c, d) ist ein einfacher Kreis.
- (e, d, a, b) ist ein einfacher Weg der Länge 3.
- (b, d, a) ist

Für den Graphen

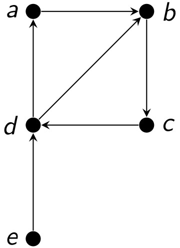


gilt:

- (e, d, b, c, d) ist ein Weg der Länge 4, aber kein einfacher Weg.
- (d, b, c, d) ist ein einfacher Kreis.
- (e, d, a, b) ist ein einfacher Weg der Länge 3.
- (b, d, a) ist kein Weg.
- (a, b, c, d, b, c, d, a) ist

Wege und Kreise: Beispiele (1/2)

Für den Graphen

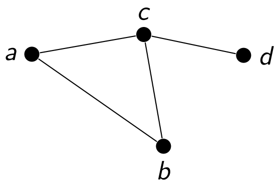


gilt:

- (e, d, b, c, d) ist ein Weg der Länge 4, aber kein einfacher Weg.
- (d, b, c, d) ist ein einfacher Kreis.
- (e, d, a, b) ist ein einfacher Weg der Länge 3.
- (b, d, a) ist kein Weg.
- (a, b, c, d, b, c, d, a) ist ein Kreis, aber kein einfacher Kreis.

Wege und Kreise: Beispiele (2/2)

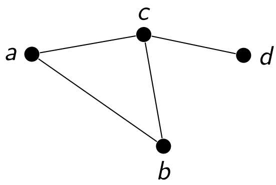
Für den Graphen



gilt:

- (a, b, c, a) ist

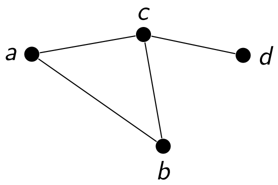
Für den Graphen



gilt:

- (a, b, c, a) ist ein einfacher Kreis.
- (c, d, c) ist

Für den Graphen

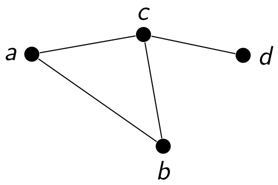


gilt:

- (a, b, c, a) ist ein einfacher Kreis.
- (c, d, c) ist ein Kreis, aber kein einfacher Kreis.
- (a, c, d) ist

Wege und Kreise: Beispiele

Für den Graphen

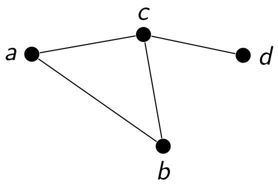


gilt:

- (a, b, c, a) ist ein einfacher Kreis.
- (c, d, c) ist ein Kreis, aber kein einfacher Kreis.
- (a, c, d) ist ein einfacher Weg der Länge 2.
- (c, b, a, c, d) ist

Wege und Kreise: Beispiele (2/2)

Für den Graphen

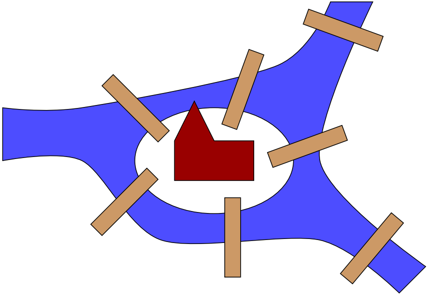


gilt:

- (a, b, c, a) ist ein einfacher Kreis.
- (c, d, c) ist ein Kreis, aber kein einfacher Kreis.
- (a, c, d) ist ein einfacher Weg der Länge 2.
- (c, b, a, c, d) ist ein Weg, aber kein einfacher Weg.

Das Königsberger Brückenproblem

In der Stadt Königsberg gab es im 18. Jahrhundert sieben Brücken über den Fluss Pregel, die die beiden Ufer und die zwei Inseln miteinander verbanden.



Gibt es einen Spaziergang, der jede der sieben Brücken genau einmal überquert und zum Ausgangspunkt zurückkehrt?

Das Königsberger Brückenproblem (2/2)

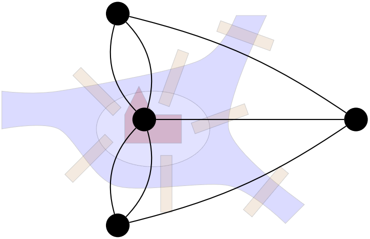
Eine Graph-theoretische Modellierung: Jedes Ufer und jede Insel wird durch einen Knoten repräsentiert. Kanten entsprechen Brücken.

Eine Graph-theoretische Modellierung: Jedes Ufer und jede Insel wird durch einen Knoten repräsentiert. Kanten entsprechen Brücken.

Die „Pregel-Skizze" wird also durch den Graphen $G_{\text{Königsberg}}$ repräsentiert:

(„Mehrfachkanten" sind ausnahmsweise erlaubt.)

$G_{\text{Königsberg}} :=$



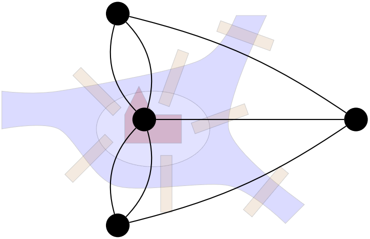
Die Frage nach dem "Spaziergang" entspricht dann der Frage:

Das Königsberger Brückenproblem (2/2)

Eine Graph-theoretische Modellierung: Jedes Ufer und jede Insel wird durch einen Knoten repräsentiert. Kanten entsprechen Brücken.

Die „Pregel-Skizze“ wird also durch den Graphen $G_{\text{Königsberg}}$ repräsentiert:
(„Mehrfachkanten“ sind ausnahmsweise erlaubt.)

$G_{\text{Königsberg}} :=$



Die Frage nach dem “Spaziergang” entspricht dann der Frage:

*Gibt es in $G_{\text{Königsberg}}$ einen „**Euler-Kreis**“,*
also einen Kreis der jede Kante genau einmal durchläuft?

Warum ist die Antwort negativ?

Routenplaner

Routenplaner: Die Länge „gewichteter“ Wege

Wie gehen Routenplaner vor, um schnellste Verbindungen zwischen einem Start und einem Ziel zu berechnen?

Die relevanten Informationen werden durch einen gerichteten Graphen repräsentiert.

Routenplaner: Die Länge „gewichteter“ Wege

Wie gehen Routenplaner vor, um schnellste Verbindungen zwischen einem Start und einem Ziel zu berechnen?

Die relevanten Informationen werden durch einen gerichteten Graphen repräsentiert.

(a) Der **gerichtete Graph**:
- Kreuzungen oder Abfahrten sind die **Knoten**,
- Straßenabschnitte, die Knoten direkt miteinander verbinden, sind die **gerichteten Kanten**.

Routenplaner: Die Länge „gewichteter“ Wege

Wie gehen Routenplaner vor, um schnellste Verbindungen zwischen einem Start und einem Ziel zu berechnen?

Die relevanten Informationen werden durch einen gerichteten Graphen repräsentiert.

(a) Der **gerichtete Graph**:
- Kreuzungen oder Abfahrten sind die **Knoten**,
- Straßenabschnitte, die Knoten direkt miteinander verbinden, sind die **gerichteten Kanten**.

(b) Jede Kante wird mit geographischer Information versehen.
- Für die **Navigation**: In welchem Stadtteil, Stadt, Bundesland oder Staat liegt der entsprechende Straßenabschnitt, welche Hausnummern kommen vor?
- Um **schnellste Verbindungen** zu berechnen:
  Weise jeder Kante eine Dauer zu, wenn der entsprechende Straßenabschnitt in Regelgeschwindigkeit gemäß den Verkehrsregeln durchfahren wird.

Routenplaner: Die Länge „gewichteter“ Wege

Wie gehen Routenplaner vor, um schnellste Verbindungen zwischen einem Start und einem Ziel zu berechnen?

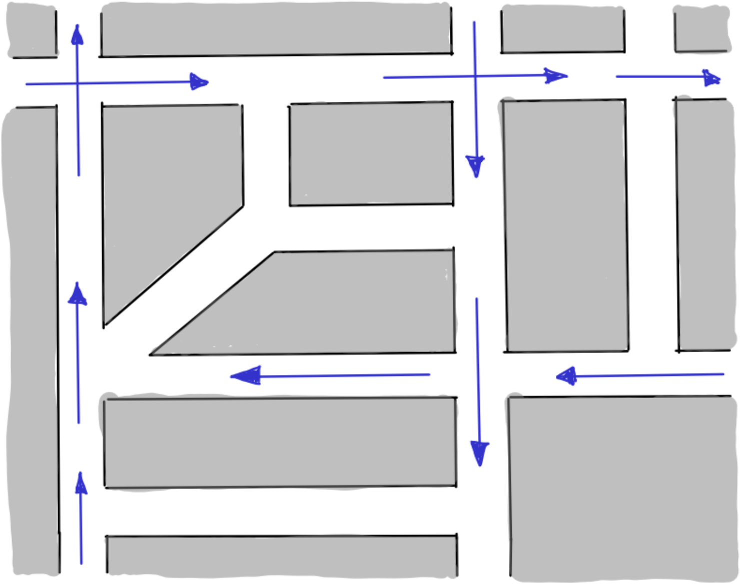
Die relevanten Informationen werden durch einen gerichteten Graphen repräsentiert.

(a) Der **gerichtete Graph**:
- Kreuzungen oder Abfahrten sind die **Knoten**,
- Straßenabschnitte, die Knoten direkt miteinander verbinden, sind die **gerichteten Kanten**.

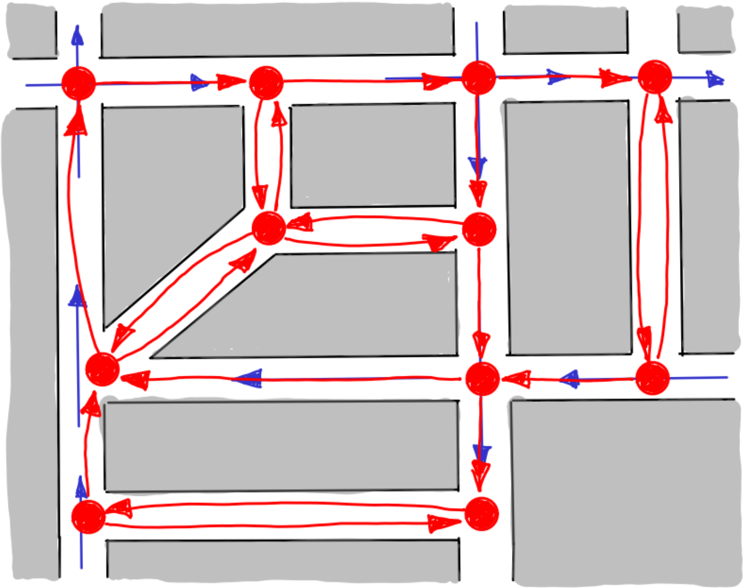
(b) Jede Kante wird mit geographischer Information versehen.
- Für die **Navigation**: In welchem Stadtteil, Stadt, Bundesland oder Staat liegt der entsprechende Straßenabschnitt, welche Hausnummern kommen vor?
- Um **schnellste Verbindungen** zu berechnen:
  Weise jeder Kante eine Dauer zu, wenn der entsprechende Straßenabschnitt in Regelgeschwindigkeit gemäß den Verkehrsregeln durchfahren wird.

Wie berechnet man den Graphen aus der Karte und wie berechnet man schnellste Verbindungen?

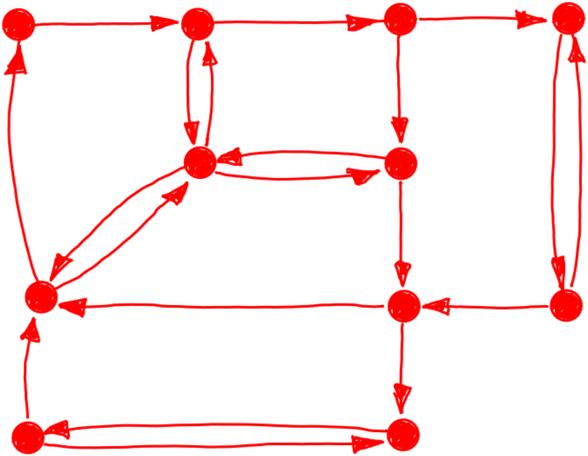
Ein Ausschnitt der Strassenkarte mit Einbahnstraßen



Der gerichtete Graph „auf“ der Karte



Der gerichtete Graph



Und wie bestimmt man schnellste Verbindungen?

Was genau ist das zugrunde liegende Problem?

DEFINITION

(a) Für den gerichteten Graphen $G = (V, E)$ weist eine Funktion

$$\text{länge} : E \to \mathbb{R}_{\geqslant 0}$$

jeder Kante eine nicht-negative Länge zu.

Und wie bestimmt man schnellste Verbindungen?

Was genau ist das zugrunde liegende Problem?

DEFINITION

(a) Für den gerichteten Graphen $G = (V, E)$ weist eine Funktion

$$\text{länge} : E \to \mathbb{R}_{\geqslant 0}$$

jeder Kante eine nicht-negative Länge zu.

(b) Für einen „Start"-Knoten $s \in V$ und einen „Ziel"-Knoten $t \in V$ bestimme einen Weg $W = s \to v_1 \to \cdots \to v_k \to t$ von s nach t, so dass Länge(W) **kleinstmöglich** ist. Es ist

$$\text{Länge}(W) = \text{länge}(s, v_1) + \text{länge}(v_1, v_2) + \cdots + \text{länge}(v_k, t).$$

Und wie bestimmt man schnellste Verbindungen?

Was genau ist das zugrunde liegende Problem?

DEFINITION

(a) Für den gerichteten Graphen $G = (V, E)$ weist eine Funktion

$$\text{länge} : E \to \mathbb{R}_{\geqslant 0}$$

jeder Kante eine nicht-negative Länge zu.

(b) Für einen „Start"-Knoten $s \in V$ und einen „Ziel"-Knoten $t \in V$ bestimme einen Weg $W = s \to v_1 \to \cdots \to v_k \to t$ von s nach t, so dass Länge(W) **kleinstmöglich** ist. Es ist

$$\text{Länge}(W) = \text{länge}(s, v_1) + \text{länge}(v_1, v_2) + \cdots + \text{länge}(v_k, t).$$

In den Veranstaltungen „**Algorithmen und Datenstrukturen 1,2**" wird eine schnelle Lösung dieses „Kürzesten-Wege-Problems" mit

dem Algorithmus von Dijkstra

beschrieben. **:-)))**

Graphen: Wichtige Anwendungsbeispiele

- Wir haben gerade die in einer **Straßenkarte** relevante Information durch einen gerichteten Graphen (mit Zusatzinformation) repräsentiert. In ähnlicher Weise lassen sich
  - das Schienennetz der deutschen Bahn oder
  - städtische S- und U-Bahn Netze veranschaulichen.
- In einem **Computer-Netzwerk** werden Computer durch Knoten und Netzwerkverbindungen durch ungerichtete Kanten repräsentiert.
- Im „**Webgraphen**“ repräsentieren die Knoten Webseiten und die gerichteten Kanten Hyperlinks.
- Um **soziale Netzwerke** zu modellieren, werden Knoten für die Akteure und Kanten für paarweise Beziehungen zwischen den Akteuren eingesetzt.
  - Bestimme Cluster von Akteuren oder
  - die wichtigen Akteure oder ...
- **Binäre Entscheidungsgraphen** (BDD's) werden in der technischen Informatik als Datenstruktur für die kompakte Darstellung und effiziente Handhabung boolescher Funktionen eingesetzt. (Siehe z.B. die Vorlesung **Rechnertechnologie und kombinatorische Schaltungen**.)

Zusammenhang

Zusammenhang

DEFINITION

Sei $G = (V, E)$ ein Graph und $v \in V$ ein Knoten.

(a) Wenn G **ungerichtet** ist, dann besteht die **Zusammenhangskomponente** von v aus allen, von v aus durch einen Weg erreichbaren Knoten.

Der Graph G heißt **zusammenhängend**, wenn die Zusammenhangskomponente irgendeines Knotens aus allen Knoten von G besteht.

Zusammenhang

DEFINITION

Sei $G = (V, E)$ ein Graph und $v \in V$ ein Knoten.

(a) Wenn G **ungerichtet** ist, dann besteht die **Zusammenhangskomponente** von v aus allen, von v aus durch einen Weg erreichbaren Knoten.

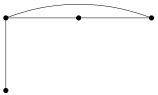
Der Graph G heißt **zusammenhängend**, wenn die Zusammenhangskomponente irgendeines Knotens aus allen Knoten von G besteht.

(b) Wenn G **gerichtet** ist, dann besteht die **starke Zusammenhangskomponente** von v aus allen Knoten w,

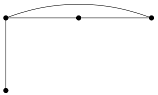
- die sowohl von v aus erreichbar sind,
- die aber auch v selbst durch einen in w beginnenden Weg erreichen.

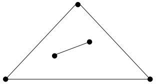
G heißt **stark zusammenhängend**, wenn die starke Zusammenhangskomponente irgendeines Knotens aus allen Knoten von G besteht.

Zusammenhang: Beispiele

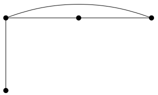
Der Graph  ist

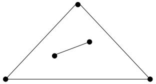
Zusammenhang: Beispiele

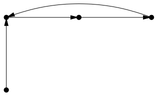
Der Graph  ist zusammenhängend,

 ist

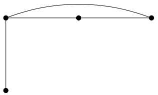
Zusammenhang: Beispiele

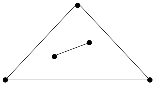
Der Graph  ist zusammenhängend,

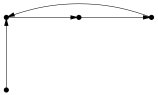
 ist nicht zusammenhängend und besteht aus zwei Zusammenhangskomponenten.

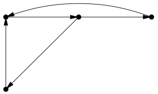
Der Graph  ist

Zusammenhang: Beispiele

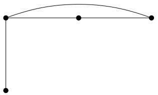
Der Graph  ist zusammenhängend,

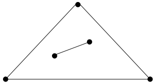
 ist nicht zusammenhängend und besteht aus zwei Zusammenhangskomponenten.

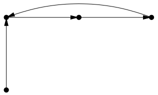
Der Graph  ist nicht stark zusammenhängend, da es z.B. keinen Weg vom Knoten links oben zum Knoten links unten gibt.

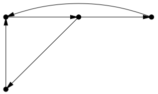
 ist

Zusammenhang: Beispiele

Der Graph  ist zusammenhängend,

 ist nicht zusammenhängend und besteht aus zwei Zusammenhangskomponenten.

Der Graph  ist nicht stark zusammenhängend, da es z.B. keinen Weg vom Knoten links oben zum Knoten links unten gibt.

 ist stark zusammenhängend.

Zuordnungsprobleme

Wer mag wen?

In einem Tennisverein sollen die Vereinsmitglieder für ein Turnier zu Doppelpaarungen zusammengestellt werden. Dabei möchte man jeweils nur befreundete Personen als „Doppel" zusammen spielen lassen.

Wir modellieren die Aufgabenstellung durch den ungerichteten Graphen $G_T := (V_T, E_T)$ mit

$$V_T \quad :=$$

Wer mag wen?

In einem Tennisverein sollen die Vereinsmitglieder für ein Turnier zu Doppelpaarungen zusammengestellt werden. Dabei möchte man jeweils nur befreundete Personen als „Doppel" zusammen spielen lassen.

Wir modellieren die Aufgabenstellung durch den ungerichteten Graphen $G_T := (V_T, E_T)$ mit

$$\begin{aligned} V_T &:= \{x \,:\, x \text{ ist ein Vereinsmitglied}\} \\ E_T &:= \end{aligned}$$

Wer mag wen?

In einem Tennisverein sollen die Vereinsmitglieder für ein Turnier zu Doppelpaarungen zusammengestellt werden. Dabei möchte man jeweils nur befreundete Personen als „Doppel“ zusammen spielen lassen.

Wir modellieren die Aufgabenstellung durch den ungerichteten Graphen $G_T := (V_T, E_T)$ mit

$$\begin{aligned} V_T &:= \{x \ : \ x \text{ ist ein Vereinsmitglied}\} \\ E_T &:= \{\{x, y\} \ : \ x \text{ und } y \text{ sind befreundete Vereinsmitglieder}\}. \end{aligned}$$

Wer mag wen?

In einem Tennisverein sollen die Vereinsmitglieder für ein Turnier zu Doppelpaarungen zusammengestellt werden. Dabei möchte man jeweils nur befreundete Personen als „Doppel" zusammen spielen lassen.

Wir modellieren die Aufgabenstellung durch den ungerichteten Graphen $G_T := (V_T, E_T)$ mit

$$\begin{aligned} V_T &:= \{x \,:\, x \text{ ist ein Vereinsmitglied}\} \\ E_T &:= \{\{x, y\} \,:\, x \text{ und } y \text{ sind befreundete Vereinsmitglieder}\}. \end{aligned}$$

Das Ziel: Finde eine größtmögliche Anzahl von Doppelpaarungen.

D.h.: Finde eine möglichst große Menge $E' \subseteq E_T$, so dass kein Vereinsmitglied Endpunkt von mehr als einer Kante aus E' ist.

Wer kann was?

Eine Gruppe unterschiedlich ausgebildeter Piloten soll so auf Flugzeuge verteilt werden, dass jeder das ihm zugeteilte Flugzeug fliegen kann.

Auch hier modellieren wir die Fragestellung durch einen ungerichteten Graphen $G_F := (V_F, E_F)$ mit

$$V_F \quad :=$$

Wer kann was?

Eine Gruppe unterschiedlich ausgebildeter Piloten soll so auf Flugzeuge verteilt werden, dass jeder das ihm zugeteilte Flugzeug fliegen kann.

Auch hier modellieren wir die Fragestellung durch einen ungerichteten Graphen $G_F := (V_F, E_F)$ mit

$$\begin{aligned} V_F &:= \{x : x \text{ ist ein Pilot}\} \cup \{y : y \text{ ist ein Flugzeug}\}, \\ E_F &:= \end{aligned}$$

Wer kann was?

Eine Gruppe unterschiedlich ausgebildeter Piloten soll so auf Flugzeuge verteilt werden, dass jeder das ihm zugeteilte Flugzeug fliegen kann.

Auch hier modellieren wir die Fragestellung durch einen ungerichteten Graphen $G_F := (V_F, E_F)$ mit

$$\begin{aligned} V_F &:= \{x \,:\, x \text{ ist ein Pilot}\} \cup \{y \,:\, y \text{ ist ein Flugzeug}\}, \\ E_F &:= \{\{x, y\} \,:\, \text{Pilot } x \text{ kann Flugzeug } y \text{ fliegen }\}. \end{aligned}$$

Wer kann was?

Eine Gruppe unterschiedlich ausgebildeter Piloten soll so auf Flugzeuge verteilt werden, dass jeder das ihm zugeteilte Flugzeug fliegen kann.

Auch hier modellieren wir die Fragestellung durch einen ungerichteten Graphen $G_F := (V_F, E_F)$ mit

$$\begin{aligned} V_F &:= \{x \,:\, x \text{ ist ein Pilot}\} \cup \{y \,:\, y \text{ ist ein Flugzeug}\}, \\ E_F &:= \{\{x, y\} \,:\, \text{Pilot } x \text{ kann Flugzeug } y \text{ fliegen }\}. \end{aligned}$$

Das Ziel: Stelle einen Flugplan auf, so dass jeder Pilot das ihm zugeteilte Flugzeug fliegen kann.

D.h.: Finde eine möglichst große Menge $E' \subseteq E_F$,
so dass kein Element aus V_F Endpunkt von mehr als einer Kante in E' ist.

DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph.

(a) Wir nennen eine Kantenmenge $E' \subseteq E$ ein **Matching** (bzw. eine Menge unabhängiger Kanten),

falls kein Knoten aus V Endpunkt von mehr als einer Kante aus E' ist.

(b) Ein Matching M heißt genau dann **perfekt**, wenn jeder Knoten aus V Endpunkt einer Kante von M ist.

Das Matching Problem (1/2)

DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph.

(a) Wir nennen eine Kantenmenge $E' \subseteq E$ ein **Matching** (bzw. eine Menge unabhängiger Kanten),

falls kein Knoten aus V Endpunkt von mehr als einer Kante aus E' ist.

(b) Ein Matching M heißt genau dann **perfekt**, wenn jeder Knoten aus V Endpunkt einer Kante von M ist.

Typischerweise, wie auch in den beiden Beispielen, möchte man ein **möglichst großes Matching** bestimmen.

Ein perfektes Matching ist größtmöglich.

Das Matching Problem (2/2)

Manchmal sind die Kanten auch mit Gewichten markiert, um auszudrücken

- wie befreundet zwei Mitglieder sind
- oder wie gut sich ein Pilot mit einem Flugzeug auskennt.

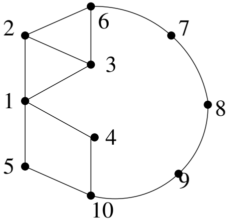
In diesem Fall möchte man ein **möglichst schweres Matching** bestimmen.

In der Wahlpflichtveranstaltung „**Approximationsalgorithmen**“
(ab dem 4. Semester) wird gezeigt, dass ein schwerstes Matching effizient bestimmt werden kann. **:-)))**

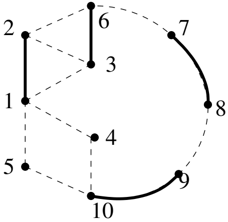
Das Matching Problem: Beispiele

In einem Tennisverein mit 10 Mitgliedern und dem „Freundschaftsgraph“

$G_T \quad =$



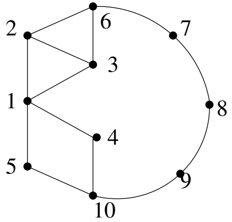
sind z.B. die folgenden beiden Kantenmengen Matchings:



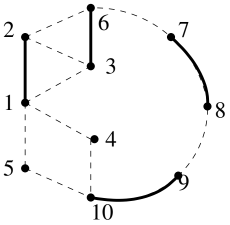
und

Das Matching Problem: Beispiele

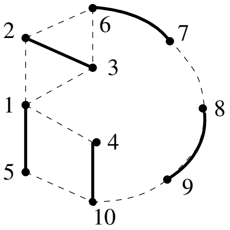
In einem Tennisverein mit 10 Mitgliedern und dem „Freundschaftsgraph“

$G_T \quad =$ 

sind z.B. die folgenden beiden Kantenmengen Matchings:



und



Schwierige Probleme: Hamilton-Kreise

Hamilton-Kreise und Hamilton-Wege

DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ein ungerichteter) Graph.

(a) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Weg**, wenn jeder Knoten aus V genau einmal in W vorkommt.

(b) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein Hamilton-Weg ist und $v_\ell = v_0$ gilt.

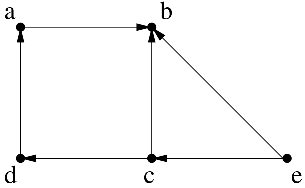
DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ein ungerichteter) Graph.

(a) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Weg**, wenn jeder Knoten aus V genau einmal in W vorkommt.

(b) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein Hamilton-Weg ist und $v_\ell = v_0$ gilt.

Der Graph G



hat einen Hamilton-Weg, nämlich

Hamilton-Kreise und Hamilton-Wege

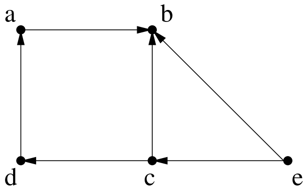
DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ein ungerichteter) Graph.

(a) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Weg**, wenn jeder Knoten aus V genau einmal in W vorkommt.

(b) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein Hamilton-Weg ist und $v_\ell = v_0$ gilt.

Der Graph G



hat einen Hamilton-Weg, nämlich $(\mathrm{e}, \mathrm{c}, \mathrm{d}, \mathrm{a}, \mathrm{b})$, aber keinen Hamilton-Kreis, da

Hamilton-Kreise und Hamilton-Wege

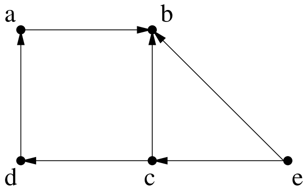
DEFINITION

Sei $G = (V, E)$ ein (gerichteter oder ein ungerichteter) Graph.

(a) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Weg**, wenn jeder Knoten aus V genau einmal in W vorkommt.

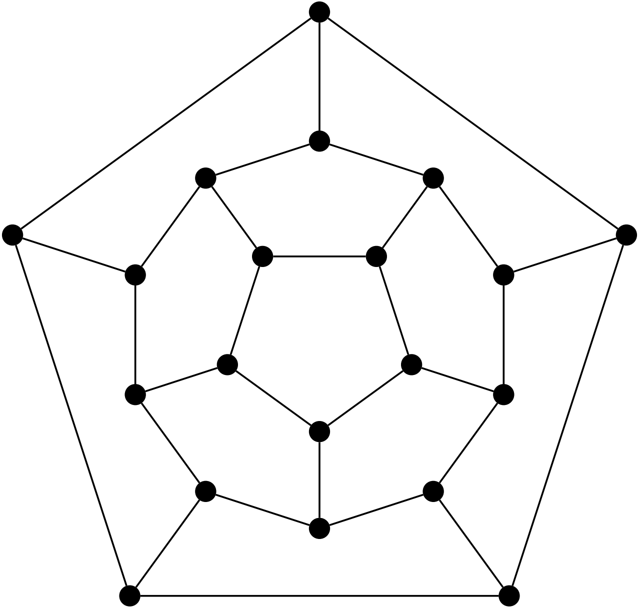
(b) Ein Weg $W = (v_0, \ldots, v_\ell)$ heißt **Hamilton-Kreis**, wenn $(v_0, \ldots, v_{\ell-1})$ ein Hamilton-Weg ist und $v_\ell = v_0$ gilt.

Der Graph G



hat einen Hamilton-Weg, nämlich $(\mathrm{e}, \mathrm{c}, \mathrm{d}, \mathrm{a}, \mathrm{b})$, aber keinen Hamilton-Kreis, da $\text{Aus-Grad}_G(b) = 0$ ist.

Wo ist der Hamilton-Kreis?



Hamilton-Kreise: Wie schwierig?

Wir erhalten einen gerichteten oder ungerichteten Graphen mit mehreren Tausenden von Knoten: Wir sollen noch nicht einmal einen Hamilton-Kreis bestimmen, sondern nur die Frage beantworten, ob es einen Hamilton-Kreis gibt.

Trauen Sie sich zu ein Programm zu schreiben, dass diese Frage zu ihren Lebzeiten beantwortet?

Hamilton-Kreise: Wie schwierig?

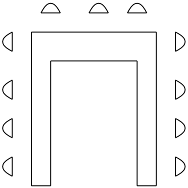
Wir erhalten einen gerichteten oder ungerichteten Graphen mit mehreren Tausenden von Knoten: Wir sollen noch nicht einmal einen Hamilton-Kreis bestimmen, sondern nur die Frage beantworten, ob es einen Hamilton-Kreis gibt.

Trauen Sie sich zu ein Programm zu schreiben, dass diese Frage zu ihren Lebzeiten beantwortet?

In der Veranstaltung „**Algorithmen und Datenstrukturen 2**“ wird gezeigt, dass auch diese Frage, wie schon das Erfüllbarkeitsproblem KNF-SAT, $\mathtt{NP}$-vollständig ist. **:-(((**

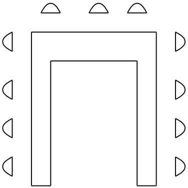
Euler-Kreise, falls vorhanden, lassen sich hingegen effizient bestimmen. (Ein Euler-Kreis durchläuft alle Kanten genau einmal.) **:-)))**

Die Gäste einer Familienfeier sollen so an einer hufeisenförmigen Tafel



platziert werden, dass niemand neben jemanden sitzt, den sie/er nicht leiden kann.

Die Gäste einer Familienfeier sollen so an einer hufeisenförmigen Tafel



platziert werden, dass niemand neben jemanden sitzt, den sie/er nicht leiden kann.

1. Stelle den **Konfliktgraphen** $G = (V, E)$ auf, wobei

$$V := \{x : \text{Person } x \text{ soll zur Feier kommen}\} \text{ und}$$

$$E := \left\{ \{x, y\} : \begin{array}{l} \text{Person } x \text{ kann Person } y \text{ nicht leiden oder} \\ \text{Person } y \text{ kann Person } x \text{ nicht leiden} \end{array} \right\}$$

d.h. Kanten im Konfliktgraphen zeigen auf, wer im Konflikt mit wem steht.

Sitzordnung: Wer neben wem? (2/2)

2. Bilde das Komplement des Konfliktgraphen,
   d.h. betrachte den **Freundschaftsgraphen** $\tilde{G} = (\tilde{V}, \tilde{E})$ mit

$$\begin{aligned} \tilde{V} &:= V \quad \text{und} \\ \tilde{E} &:= \big\{ \{x,y\} \,:\, x,y \in V,\, x \neq y,\, \{x,y\} \notin E \big\}. \end{aligned}$$

   Kanten in $\tilde{G}$ zeigen an, wer prinzipiell neben wem platziert werden könnte.

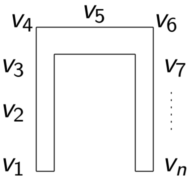
Sitzordnung: Wer neben wem? (2/2)

2. Bilde das Komplement des Konfliktgraphen,
   d.h. betrachte den **Freundschaftsgraphen** $\tilde{G} = (\tilde{V}, \tilde{E})$ mit

$$\begin{aligned} \tilde{V} &:= V \quad \text{und} \\ \tilde{E} &:= \big\{ \{x,y\} \;:\; x,y \in V,\, x \neq y,\, \{x,y\} \notin E \big\}. \end{aligned}$$

   Kanten in $\tilde{G}$ zeigen an, wer prinzipiell neben wem platziert werden könnte.

3. Suche einen Hamilton-Weg in $\tilde{G}$.
   Wenn $(v_1, \ldots, v_n)$ (mit $n = |\tilde{V}|$) ein Hamilton-Weg in $\tilde{G}$ ist, dann kann man die Sitzordnung folgendermaßen festlegen:



Schwierige Probleme: Das Färbungsproblem

Färbung von Knoten

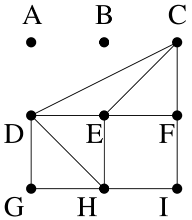
DEFINITION

Sei $G = (V, E)$ ein ungerichteter Graph und F eine Menge.

- Eine Funktion $m\colon V \to F$ heißt eine **Färbung** von G, wenn für jede Kante $\{x, y\} \in E$ gilt: $m(x) \neq m(y)$.
- Eine Färbung m von G heißt eine **k-Färbung** von G, wenn F genau k Elemente besitzt.
- G heißt **k-färbbar**, wenn G eine k-Färbung besitzt. Ist G k-färbbar, aber nicht $(k-1)$-färbbar, dann heißt $\chi(G) := k$ die **chromatische Zahl** von G.

Wieviele Farben? (1/2)

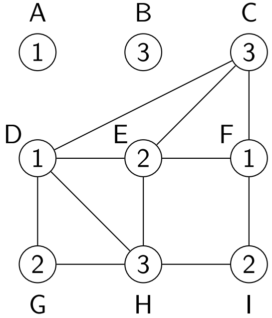
Eine Familienfeier mit Gästen A, B, C, D, E, F, G, H, I und dem Konfliktgraphen:



Bestimme die chromatische Zahl!

Wieviele Farben? (2/2)

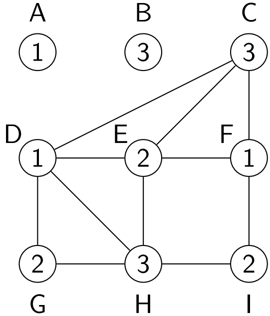
Eine 3-Färbung ist möglich:



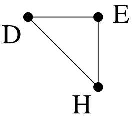
Drei Farben sind auch notwendig, weil

Wieviele Farben? (2/2)

Eine 3-Färbung ist möglich:



Drei Farben sind auch notwendig, weil der Konfliktgraph ein Dreieck, z.B.



als Teilgraph enthält.

Das 4-Farben Problem

Wie viele Farben sind nötig, um jede Landkarte so einzufärben, dass zwei Staaten mit gemeinsamer Grenze durch unterschiedliche Farben dargestellt werden?

1976 wurde bewiesen, dass vier Farben ausreichen.
Der Beweis basiert auf einer Fallunterscheidung mit mehr als 1000 Fällen, die mit Hilfe eines Computerprogramms analysiert wurden.

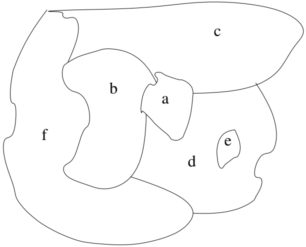
Wir modellieren eine Landkarte durch einen ungerichteten Graphen,

- dessen Knoten die Staaten repräsentieren,
- und bei dem zwei Staaten genau dann durch eine Kante miteinander verbunden sind, wenn sie eine gemeinsame Grenze besitzen.

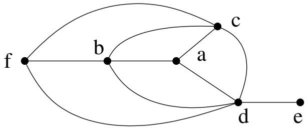
Finde eine Färbung mit möglichst wenigen Farben.

Färbung von Landkarten und planare Graphen (1/2)

Wir betrachten eine kleine Landkarte



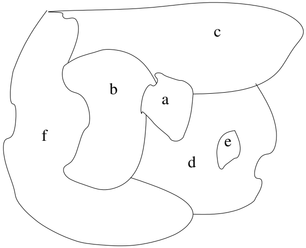
und ihren Konfliktgraphen:



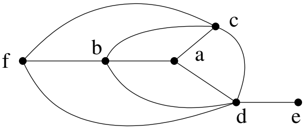
Knoten $\widehat{=}$ Staaten
Kanten $\widehat{=}$ Staaten mit gemeinsamer Grenze

Färbung von Landkarten und planare Graphen (1/2)

Wir betrachten eine kleine Landkarte



und ihren Konfliktgraphen:



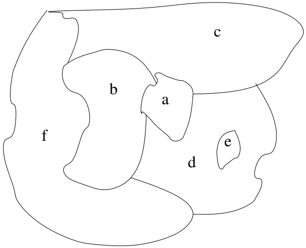
Knoten $\widehat{=}$ Staaten
Kanten $\widehat{=}$ Staaten mit gemeinsamer Grenze

Jeder der vier Knoten a, b, c, d ist mit jedem anderen benachbart $\Longrightarrow$

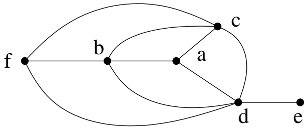
Eine Färbung muss den vier Knoten vier verschiedene Farben zuordnen. — für a, b, c, d etwa *rot*, *gelb*, *grün*, *blau*.

Färbung von Landkarten und planare Graphen (1/2)

Wir betrachten eine kleine Landkarte



und ihren Konfliktgraphen:



Knoten $\widehat{=}$ Staaten
Kanten $\widehat{=}$ Staaten mit gemeinsamer Grenze

Jeder der vier Knoten a, b, c, d ist mit jedem anderen benachbart $\Longrightarrow$

Eine Färbung muss den vier Knoten vier verschiedene Farben zuordnen. — für a, b, c, d etwa *rot*, *gelb*, *grün*, *blau*.

Da f außerdem mit b, c, d benachbart ist, muss f die Farbe *rot* erhalten; e kann jede Farbe außer *blau* erhalten.

Färbung von Landkarten und planare Graphen (2/2)

Die aus Landkarten entstehenden Konfliktgraphen sind **planar**.

DEFINITION

Ein Graph G heißt **planar**, wenn er so in die Ebene gezeichnet werden kann, dass seine Kanten sich nicht kreuzen.

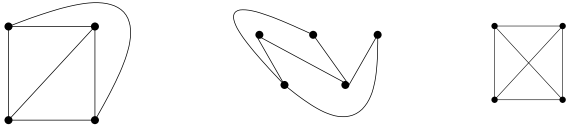
Färbung von Landkarten und planare Graphen (2/2)

Die aus Landkarten entstehenden Konfliktgraphen sind **planar**.

DEFINITION

Ein Graph G heißt **planar**, wenn er so in die Ebene gezeichnet werden kann, dass seine Kanten sich nicht kreuzen.

Beispiele für planare Graphen sind:



Der dritte Graph ist planar, da

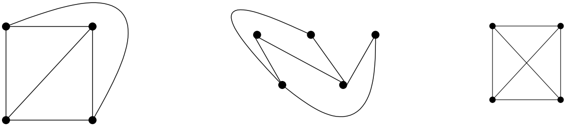
Färbung von Landkarten und planare Graphen (2/2)

Die aus Landkarten entstehenden Konfliktgraphen sind **planar**.

DEFINITION

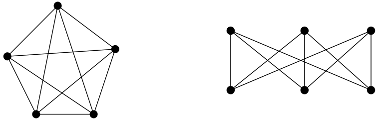
Ein Graph G heißt **planar**, wenn er so in die Ebene gezeichnet werden kann, dass seine Kanten sich nicht kreuzen.

Beispiele für planare Graphen sind:



Der dritte Graph ist planar, da er wie der erste Graph kreuzungsfrei in die Ebene gezeichnet werden kann.

Beispiele für nicht-planare Graphen sind:



Anwendungen des Färbungsproblems

Knoten	Kanten	Farbe/Markierung
Staat auf Karte	gemeinsame Grenze	Farbe
Gast auf Feier	können sich nicht leiden	Tischnummer
Vorlesung	gemeinsame Hörer	Termin
Prozess	benötigen dieselbe Ressource	Ausführungstermin

Anwendungen des Färbungsproblems

Knoten	Kanten	Farbe/Markierung
Staat auf Karte	gemeinsame Grenze	Farbe
Gast auf Feier	können sich nicht leiden	Tischnummer
Vorlesung	gemeinsame Hörer	Termin
Prozess	benötigen dieselbe Ressource	Ausführungstermin

Aber leider ist schon die Frage, ob 3 Farben ausreichen, NP-vollständig. **:-(((**

Isomorphie von Graphen

Gleichheit und Isomorphie von Graphen (1/2)

DEFINITION

$G = (V, E)$ und $G' = (V', E')$ seien zwei (gerichtete oder ungerichtete) Graphen.

(a) G und G' heißen **gleich** (kurz: $G = G'$), falls

$$V = V' \text{ und } E = E'.$$

DEFINITION

$G = (V, E)$ und $G' = (V', E')$ seien zwei (gerichtete oder ungerichtete) Graphen.

(a) G und G' heißen **gleich** (kurz: $G = G'$), falls

$$V = V' \text{ und } E = E'.$$

(b) G und G' heißen **isomorph** (kurz: $G \cong G'$, in Worten: G ist isomorph zu G'), falls es eine bijektive Abbildung $f : V \to V'$ gibt, so dass f. a. Knoten $i \in V$ und $j \in V$ gilt:

- Falls G und G' gerichtet sind:

$$(i, j) \in E \iff$$

Gleichheit und Isomorphie von Graphen (1/2)

DEFINITION

$G = (V, E)$ und $G' = (V', E')$ seien zwei (gerichtete oder ungerichtete) Graphen.

(a) G und G' heißen **gleich** (kurz: $G = G'$), falls

$$V = V' \text{ und } E = E'.$$

(b) G und G' heißen **isomorph** (kurz: $G \cong G'$, in Worten: G ist isomorph zu G'), falls es eine bijektive Abbildung $f : V \to V'$ gibt, so dass f. a. Knoten $i \in V$ und $j \in V$ gilt:

- Falls G und G' gerichtet sind:

$$(i, j) \in E \iff \big(f(i), f(j)\big) \in E'.$$

- Falls G und G' ungerichtet sind:

$$\{i, j\} \in E \iff$$

DEFINITION

$G = (V, E)$ und $G' = (V', E')$ seien zwei (gerichtete oder ungerichtete) Graphen.

(a) G und G' heißen **gleich** (kurz: $G = G'$), falls

$$V = V' \text{ und } E = E'.$$

(b) G und G' heißen **isomorph** (kurz: $G \cong G'$, in Worten: G ist isomorph zu G'), falls es eine bijektive Abbildung $f : V \to V'$ gibt, so dass f. a. Knoten $i \in V$ und $j \in V$ gilt:

- Falls G und G' gerichtet sind:

$$(i, j) \in E \iff \big(f(i), f(j)\big) \in E'.$$

- Falls G und G' ungerichtet sind:

$$\{i, j\} \in E \iff \{f(i), f(j)\} \in E'.$$

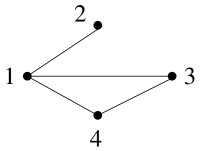
Eine solche Abbildung f wird **Isomorphismus** von G nach G' genannt.

Gleichheit und Isomorphie von Graphen (2/2)

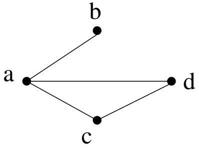
Zwei (gerichtete oder ungerichtete) Graphen G_1 und G_2 sind genau dann isomorph, wenn G_1 und G_2 gleich sind,

nachdem die Knoten von G_2 umbenannt werden.

Die beiden Graphen



und



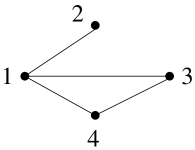
sind

Gleichheit und Isomorphie von Graphen (2/2)

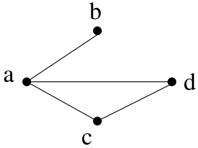
Zwei (gerichtete oder ungerichtete) Graphen G_1 und G_2 sind genau dann isomorph, wenn G_1 und G_2 gleich sind,

nachdem die Knoten von G_2 umbenannt werden.

Die beiden Graphen

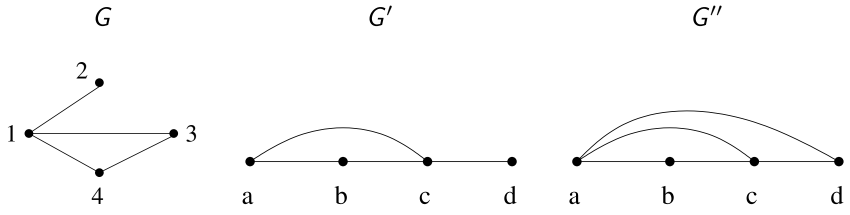


und



sind nicht gleich, da sie unterschiedliche Knotenmengen besitzen, wohl aber isomorph.

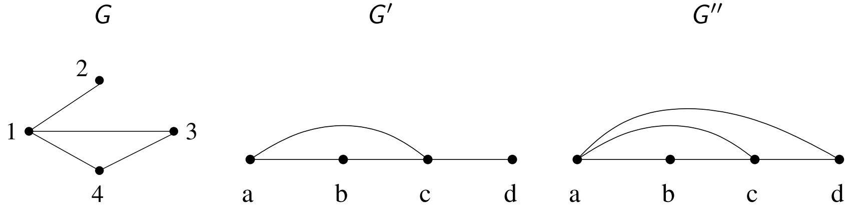
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) =$

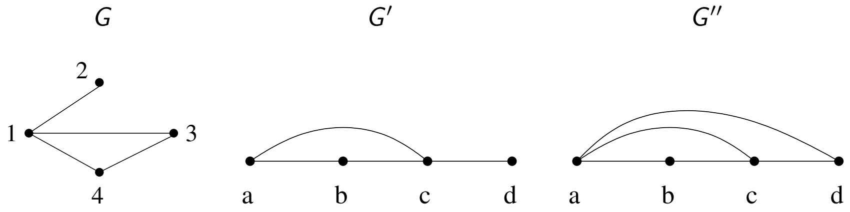
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) =$

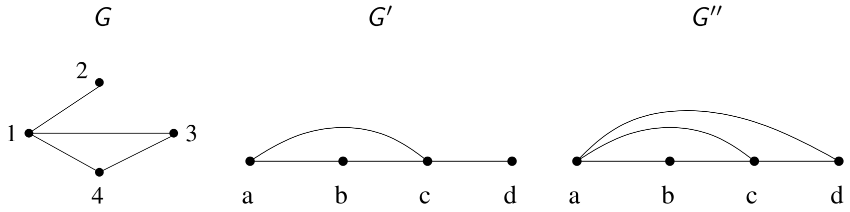
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) = d$, $f(3) =$

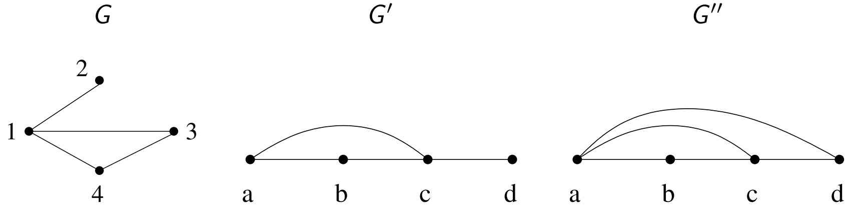
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) = d$, $f(3) = a$, $f(4) =$

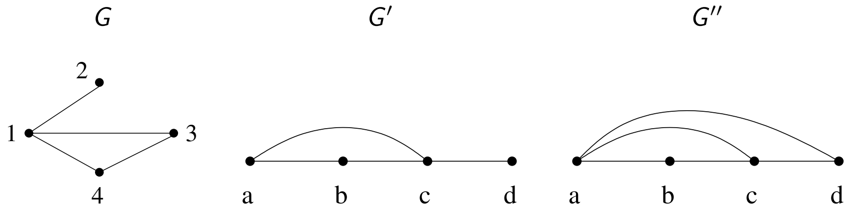
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) = d$, $f(3) = a$, $f(4) = b$ (oder

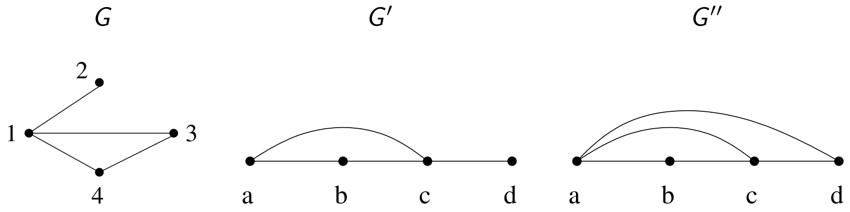
Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) = d$, $f(3) = a$, $f(4) = b$ (oder $f(3) = b$, $f(4) = a$).
- G' ist nicht isomorph zu G'' (kurz: $G' \ncong G''$),
  denn

Gleichheit und Isomorphie: Weitere Beispiele



Dann gilt:

- $G \cong G'$ via $f\colon \{1,2,3,4\} \to \{a,b,c,d\}$
  mit $f(1) = c$, $f(2) = d$, $f(3) = a$, $f(4) = b$ (oder $f(3) = b$, $f(4) = a$).
- G' ist nicht isomorph zu G'' (kurz: $G' \ncong G''$),
  denn G'' hat mehr Kanten als G'.

Wichtige Graphklassen

Vollständige Graphen und der Würfel

(a) Der **vollständige Graph** $K_n = (\{1, \ldots, n\}, E_n)$ mit n Knoten ist ein ungerichteter Graph: K_n besitzt für je zwei verschiedene Knoten eine Kante, es ist

$$E_n = \{ \{i, j\} : 1 \leqslant i \neq j \leqslant n \}.$$

Ein Graph G heißt **vollständig**, wenn es ein $n \in \mathbb{N}$ gibt mit $G \cong K_n$.

Vollständige Graphen und der Würfel

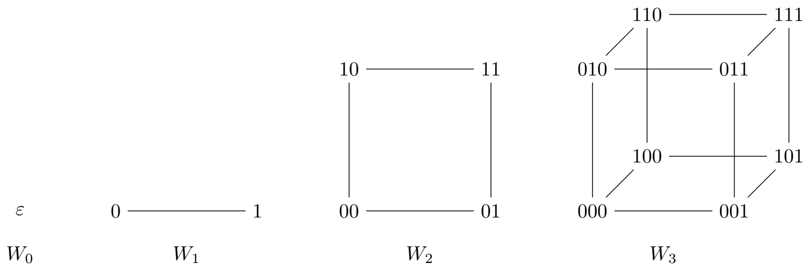
(a) Der **vollständige Graph** $K_n = (\{1, \ldots, n\}, E_n)$ mit n Knoten ist ein ungerichteter Graph: K_n besitzt für je zwei verschiedene Knoten eine Kante, es ist

$$E_n = \{ \{i, j\} : 1 \leqslant i \neq j \leqslant n \}.$$

Ein Graph G heißt **vollständig**, wenn es ein $n \in \mathbb{N}$ gibt mit $G \cong K_n$.

(b) Der **d-dimensionale Würfel $\mathbf{W_d}$** besitzt die Knotenmenge $V_d = \{0, 1\}^d$ und die Kantenmenge

$$E_d = \{ \{u, v\} : u, v \in \{0, 1\}^d, \ u, v \text{ unterscheiden sich in genau einem Bit } \}.$$



Der Würfel W_d für $d \in \{0, \ldots, 3\}$.

Planare, bipartite und kreisfreie Graphen

(c) **Planare** Graphen haben wir schon kennengelernt.

Planare, bipartite und kreisfreie Graphen

(c) **Planare** Graphen haben wir schon kennengelernt.

(d) Ein ungerichteter Graph $G = (V, E)$ heißt **bipartit**, wenn die Knotenmenge $V = V_1 \cup V_2$ so in zwei disjunkte Teilmengen V_1, V_2 zerlegt werden kann, dass alle Kanten genau einen Endpunkt in V_1 und einen Endpunkt in V_2 besitzen.

- Bipartite Graphen tauchen häufig in Zuordnungsproblemen auf.

Planare, bipartite und kreisfreie Graphen

(c) **Planare** Graphen haben wir schon kennengelernt.

(d) Ein ungerichteter Graph $G = (V, E)$ heißt **bipartit**, wenn die Knotenmenge $V = V_1 \cup V_2$ so in zwei disjunkte Teilmengen V_1, V_2 zerlegt werden kann, dass alle Kanten genau einen Endpunkt in V_1 und einen Endpunkt in V_2 besitzen.

- Bipartite Graphen tauchen häufig in Zuordnungsproblemen auf.

(e) Ein gerichteter Graph ohne Kreise heißt **azyklisch**.

- Eine Menge V von Aufgaben mit Präzedenzen $E \subseteq V \times V$ definiert einen azyklischen Graph, wenn die Aufgaben ohne Deadlock ausführbar sind.

Und wenn ein ungerichteter Graph keine Kreise besitzt?

Einfache und schwierige Probleme

Welche Probleme haben wir im Griff, welche nicht?

+ Die beherrschbaren Probleme,
  (a) das systematische Durchsuchen von Graphen (mit Hilfe der Tiefensuche),
  (b) die Bestimmung kürzester Wege,
  (c) die Berechnung von Euler-Wegen und Euler-Kreisen,
  (d) die Berechnung eines größtmöglichen Matchings
  (e) oder die Bestimmung minimaler Spannbäume.

- Die schwierigen Probleme (genau so schwierig wir KNF-SAT),
  (a) die Bestimmung von Hamilton-Kreisen oder Hamilton-Wegen, bzw.
  (b) die Bestimmung einer Färbung mit möglichst wenigen Farben.

Viel, viel mehr in den „Algorithmen und Datenstrukturen 2“.