

Algorithmische Graphentheorie

Sommersemester 2026

1. Vorlesung

Graphen: Eine Einführung

Termine & das Kleingedruckte

Vorlesung:

- Folien und Videos auf WueCampus
- Mittwochs, 10:15–11:45, Vorlesung im HS 2
Fragen: mündlich in der VL oder schriftlich im [Diskussionsforum](#).
- Dozent: Prof. Alexander Wolff
(Büro M4.01.001, [Sprechstunde](#): Mi, 14:00–15:00)

Alle Links dazu auf WueCampus!

Übungen:

- Organisation: Antonio Lauerbach
(Büro M4.01.003, [dort oder per Email erreichbar](#))
- TutorInnen:
Christopher Brandt, Duy-Khang Tran, Felix Bartunek
- Freitags, 8:30 (SE I), 10:15 (SE 8 – Physikgeb.), 12:15 (SE I)
- Erstmals schon diese Woche, **17.4.!**

Termine & das Kleingedruckte

Übungsaufgaben:

- Bearbeitung in Gruppen von max. je **zwei** TeilnehmerInnen
- Ausgabe: mittwochs via WueCampus
- Abgabe: dienstags, 14:00 Uhr, auf WueCampus (nur pdf)
Lösungen bitte mit \LaTeX schreiben! → Template
- Klausurbonus: eine Notenstufe (0,3) bei einer bestanden Klausur,
wenn 50 % der Übungsblattpunkte erreicht werden.
- Manche Übungsaufgaben müssen mit PuLP bearbeitet werden.
- Dafür gibt es am Fr, 24.4., eine Einführung in der Übung

2x Anmelden!

Melden Sie sich *sofort* bei WueStudy & WueCampus an.

- Übungseinteilung bis Mi, 15.4., 23:59, über **WueCampus**
- WueCampus auch für Kommunikation, z.B. falls eine Übungsaufgabe fehlerhaft oder missverständlich ist.
(Zum Einschreiben klicken Sie oben unter dem Kursnamen auf “Mich in diesem Kurs einschreiben”.)
- Prüfungsanmeldung bis Fr, **10.7.**, 23:59, über WueStudy

Klausuren:

- 1. Termin: Ende Juli [Anmeldung 16.04.–10.07.]
- 2. Termin: Anfang Oktober [Anmeldung 01.09.–30.09.]
- Wenn Sie sich nicht fristgerecht bei WueStudy anmelden, ist es für uns **unmöglich**, Ihre Note zu verbuchen.

Bücher zur Vorlesung



[KN]



[CLRS]

[G]

Hauptreferenz; elektronische Kopie über die Unibib erhältlich

Voraussetzungen

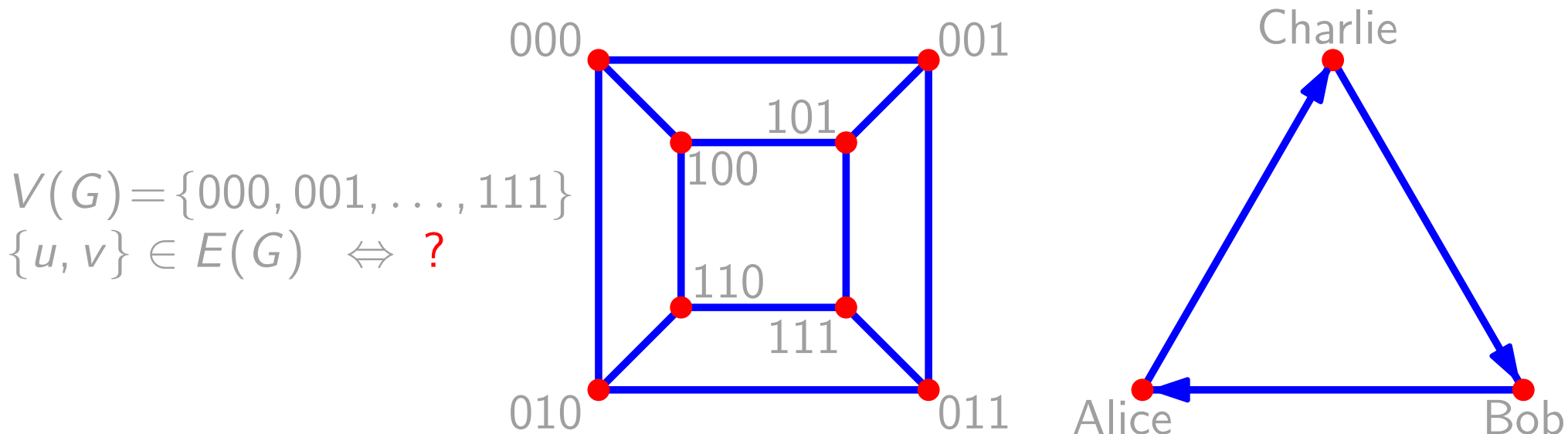
Wissen aus der Vorlesung *Algorithmen und Datenstrukturen*:

- Graphdurchlauf-Strategien
 - Breitensuche
 - Tiefensuche
- Berechnung kürzester Wege
 - Breitensuche
 - Algorithmus von Dijkstra
- Minimale Spannbäume
 - Algorithmus von Kruskal
 - Algorithmus von Jarník–Prim

Repetitorium in der
allerersten Übung

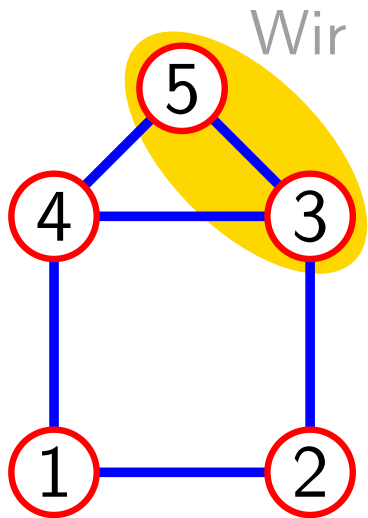
F: Was ist ein Graph?

- A₁: Ein (ungerichteter) Graph G ist ein Tupel $(V(G), E(G))$:
- $V(G)$ *Knotenmenge* und
 - $E(G) \subseteq \binom{V}{2} = \{\{u, v\} \subseteq V(G) \mid u \neq v\}$ *Kantenmenge*.



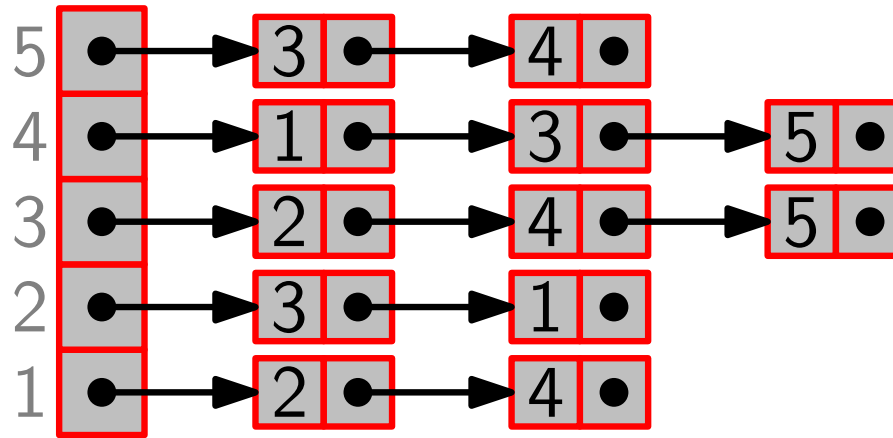
- A₂: Ein *gerichteter* Graph G ist ein Tupel $(V(G), E(G))$, wobei
- $V(G)$ *Knotenmenge* und
 - $E(G) \subseteq V(G) \times V(G) = \{(u, v) \mid u, v \in V(G), u \neq v\}$ *Kantenmenge*.

F: Wie repräsentiere ich einen Graphen?



ungerichteter
Graph

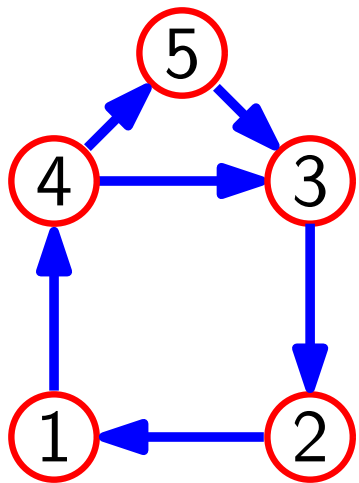
Wir sagen: Knoten 3 und 5 sind *adjazent*.



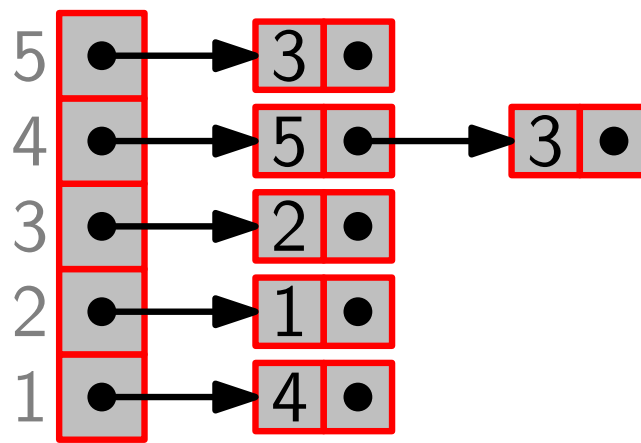
Adjazenzlisten

	1	2	3	4	5
1	0	1	0	1	0
2	1	0	1	0	0
3	0	1	0	1	1
4	1	0	1	0	1
5	0	0	1	1	0

Adjazenzmatrix



gerichteter
Graph



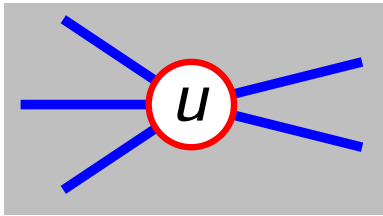
$$\text{Adj}[i] = \{j \in V(G) : (i, j) \in E(G)\}$$

	1	2	3	4	5
1	0	0	0	1	0
2	1	0	0	0	0
3	0	1	0	0	0
4	0	0	1	0	1
5	0	0	1	0	0

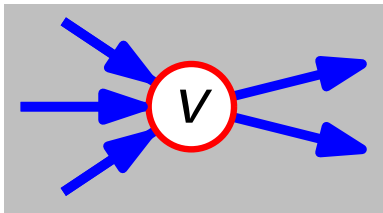
$$a_{ij} = 1 \Leftrightarrow (i, j) \in E(G)$$

Grad eines Knotens

Def.



$$\deg(u) = |\text{Adj}[u]|$$



$$\text{outdeg}(v) = |\text{Adj}[v]|$$

$$\text{indeg}(v) = |\{u \in V(G) : (u, v) \in E(G)\}|$$

Beob.

Sei G ein ungerichteter Graph.

Dann ist die Summe aller Knotengrade $= 2 \cdot |E(G)|$.

Beweis.

Technik des *zweifachen Abzählens*:

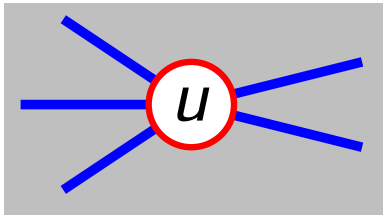
Zähle alle Knoten-Kanten-Inzidenzen.

Eine Kante ist *inzident* zu ihren Endknoten.

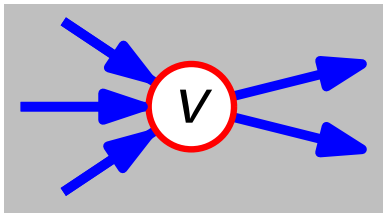
Ein Knoten ist *inzident* zu allen Kanten, deren Endknoten er ist.

Grad eines Knotens

Def.



$$\deg(u) = |\text{Adj}[u]|$$



$$\text{outdeg}(v) = |\text{Adj}[v]|$$

$$\text{indeg}(v) = |\{u \in V(G) : (u, v) \in E(G)\}|$$

Beob.

Sei G ein ungerichteter Graph.

Dann ist die Summe aller Knotengrade $= 2 \cdot |E(G)|$.

Beweis.

Technik des *zweifachen Abzählens*:

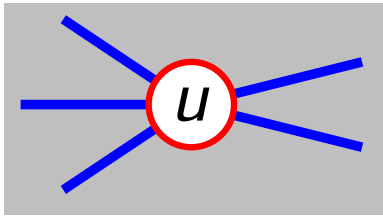
Zähle alle Knoten-Kanten-Inzidenzen.

Aus Sicht der Knoten: $\sum_{v \in V(G)} \deg(v)$

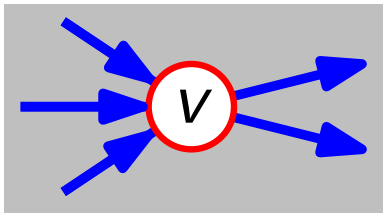
Aus Sicht der Kanten: $\sum_{e \in E(G)} 2 = 2 \cdot |E(G)|$ also gleich!

Grad eines Knotens

Def.



$$\deg(u) = |\text{Adj}[u]|$$



$$\text{outdeg}(v) = |\text{Adj}[v]|$$

$$\text{indeg}(v) = |\{u \in V(G) : (u, v) \in E(G)\}|$$

Beob.

Sei G ein ungerichteter Graph.

Dann ist die Summe aller Knotengrade $= 2 \cdot |E(G)|$.

Sätze.

Die Anzahl der Knoten ungeraden Grades ist gerade.

$$V(G) = V_{\text{ger}}(G) \cup V_{\text{ung}}(G)$$

Beweis.

$$2 \cdot |E(G)| = \sum_{v \in V_{\text{ger}}(G)} \deg(v) + \sum_{v \in V_{\text{ung}}(G)} \deg(v)$$

gerade!

gerade!

\Rightarrow *gerade!*

$$\sum_{v \in V_{\text{ung}}(G)} \deg(v) \text{ gerade} \Rightarrow |V_{\text{ung}}(G)| \text{ gerade!} \quad \square$$