Algorithmen und Wahrscheinlichkeit

Woche 1

Ilya Maier

Heute

- Motivation
- Organisatorisches
- Zusammenhang

Motivation

- Student perspective:
- Leichter **Bonus**
  - 2p. code expert
  - 2p. minitest
  - 2p. serie / peergrading
  - 80% insgesamt für 0.25
  - **Aktive** Mitarbeit = 50-75% der Prüfung

Prüfung

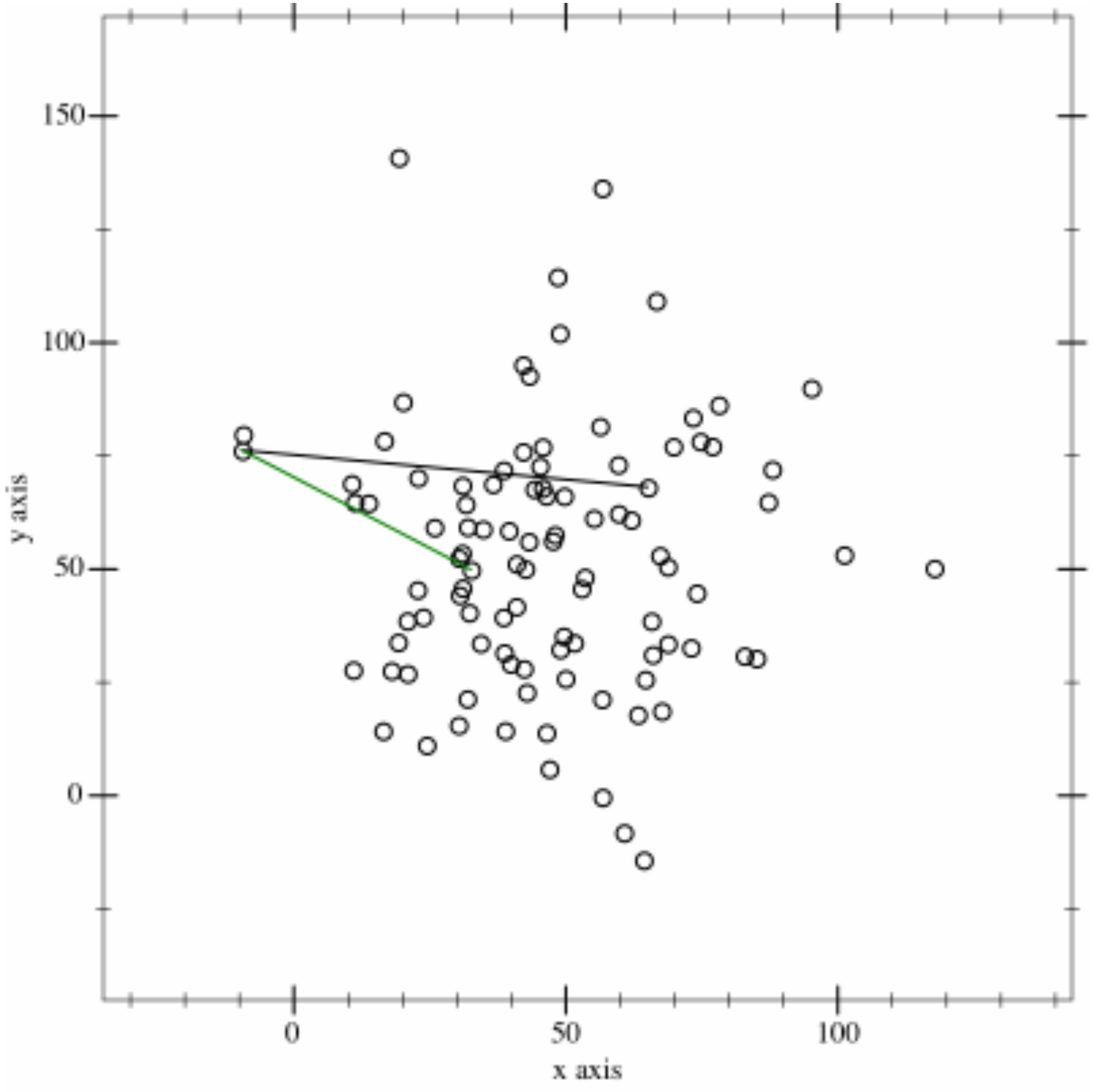
- 50% multiple choice / short answers
- 25% schriftlich
- 25% code expert

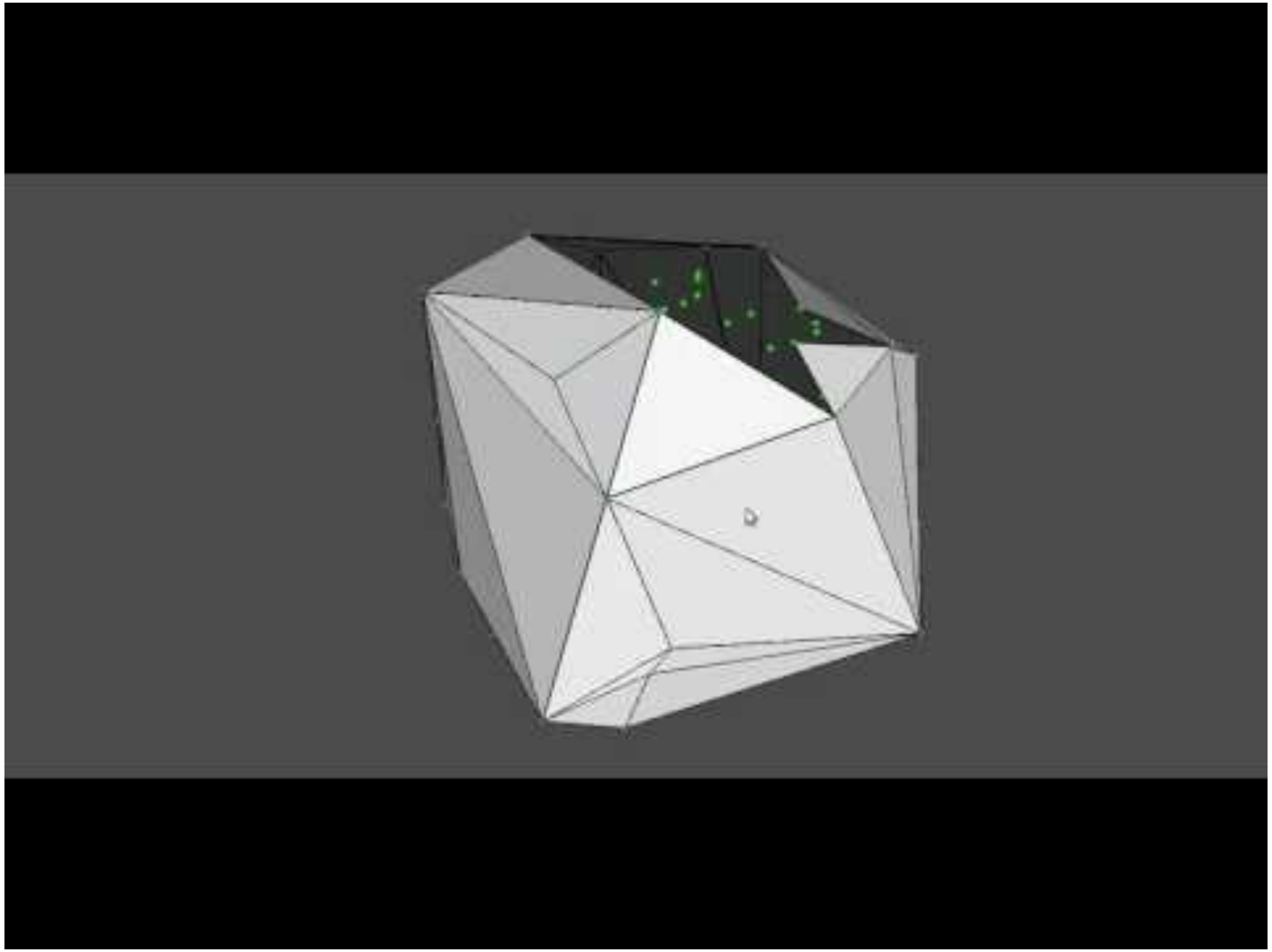
Motivation

- Student perspective:
    - Ana, PProg, DDCA -> man kann nicht wirklich was skippen
    - AnW: 3 Teile, man kann “neu” in das Fach einsteigen
      Rest im Sommer nachholen
    - Eine Woche frei im April, Feiertage im Mai, 2.5 Monate Zeit bis zu den Prüfungen -> entspannteres Semester, mehr Zeit zum lernen

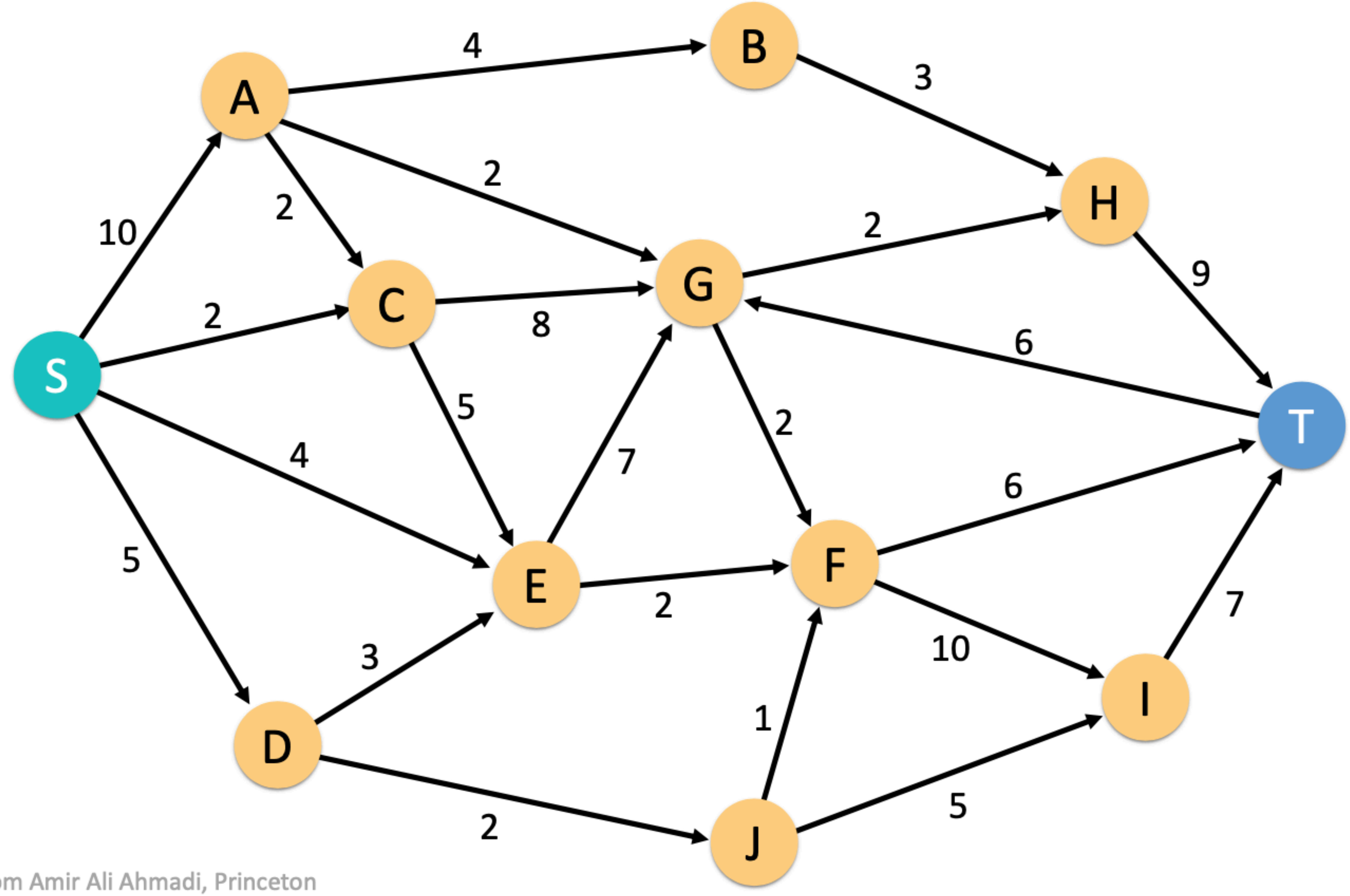
Motivation

- Inhalt perspective:
  - **Einfache** und coole Algorithmen
  - Gut und verständlich geschriebenes Skript



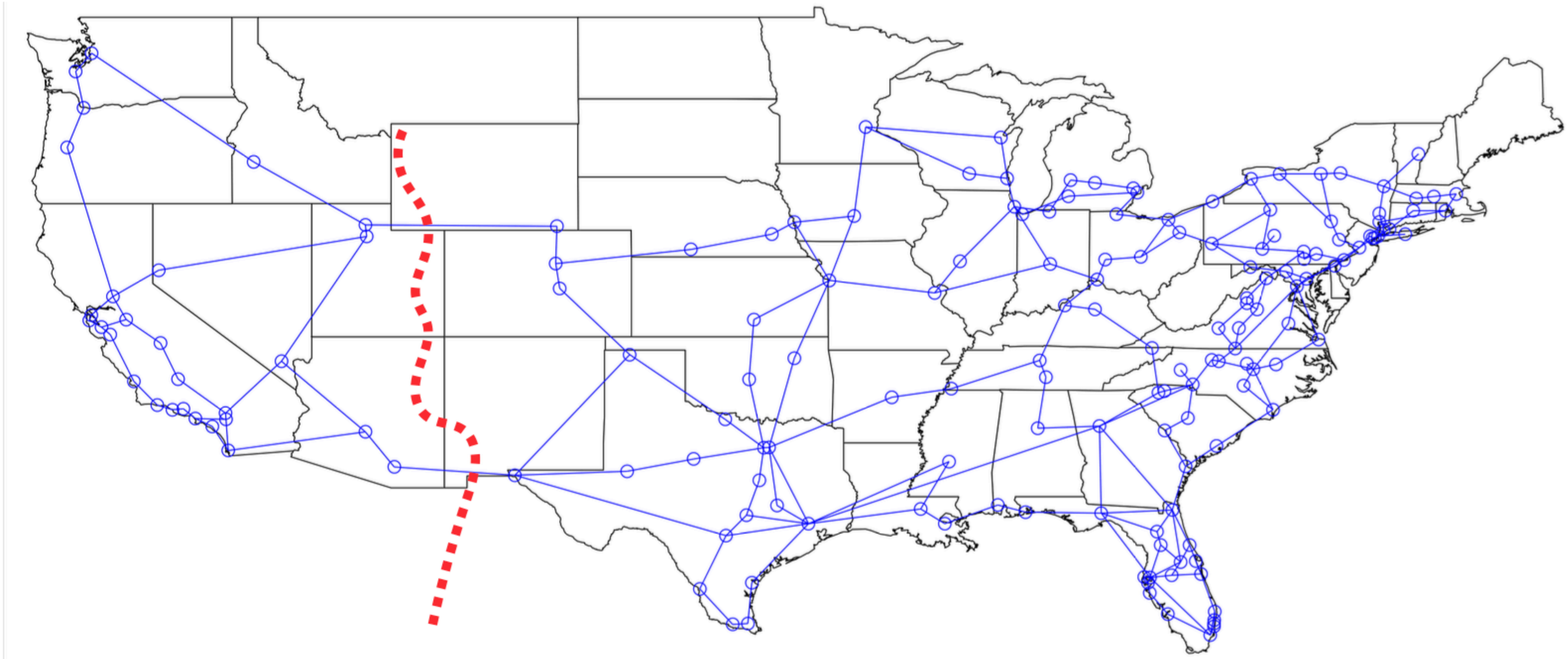


Maximum traffic from S to T?



Example adapted from Amir Ali Ahmadi, Princeton

The Internet is today's critical infrastructure!



Network Vulnerability to Single, Multiple, and Probabilistic Physical Attacks. Agarwal et. al.

Organisatorisches

- Jede Woche:
    - Code Expert ab 18 Uhr am Do. -> Abgabe bis nächsten Do um 10
- Alle 2 Wochen alternierend:
    - Minitest in der ÜS, nach der ÜS Serie - Abgabe bis nächsten Do um 10
    - PeerGrading ab 18 Uhr am Do. -> Abgabe bis So -> Korrektur bis Do
- Erster Minitest und Serie **nächsten Do**, den 29. Februar

Organisatorisches

- Abgaben auf Moodle, Korrekturen auch
- Fragen: imaier@ethz.ch
- Webseite: ilyamaier.github.io

Theorie Recap

k-Zusammenhang

Ein Graph $G = (V, E)$ ist **zusammenhängend** $\iff \forall u, v \in V, u \neq v : \exists u, v$**-Pfad** in G

<u>Knoten</u>

$X \subseteq V$

k-zusammenhängend
1) $|V| \geq k + 1$
2) $\forall X \subseteq V : |X| < k \implies G[V \backslash X]$ zusammenhängend

Satz von Menger
G k-zusammenhängend
$\iff \forall u, v \in V, u \neq v : \exists k$ intern-knotendisjunkte u, v-Pfade

<u>Kanten</u>

$X \subseteq E$

k-kanten-zusammenhängend
$\forall X \subseteq E : |X| < k \implies (V, E \backslash X)$ zusammenhängend

Satz von Menger
G k-kanten-zusammenhängend
$\iff \forall u, v \in V, u \neq v : \exists k$ intern-kantendisjunkte u, v-Pfade

$\exists v \in V : \deg(v) < k \implies G$ ist nicht k-zusammenhängend

Knotenzusammenhang $\leq$ Kantenzusammenhang $\leq$ minimaler Grad

2-Zusammenhang

Für einen zusammenhängenden Graphen $G = (V, E)$**:**

Knoten

$v \in V$ ist ein **Artikulationsknoten (AK)**

$\iff G - v$ ist **nicht zusammenhängend**

Kanten

$e \in E$ ist eine **Brücke**

$\iff G - e$ ist **nicht zusammenhängend**

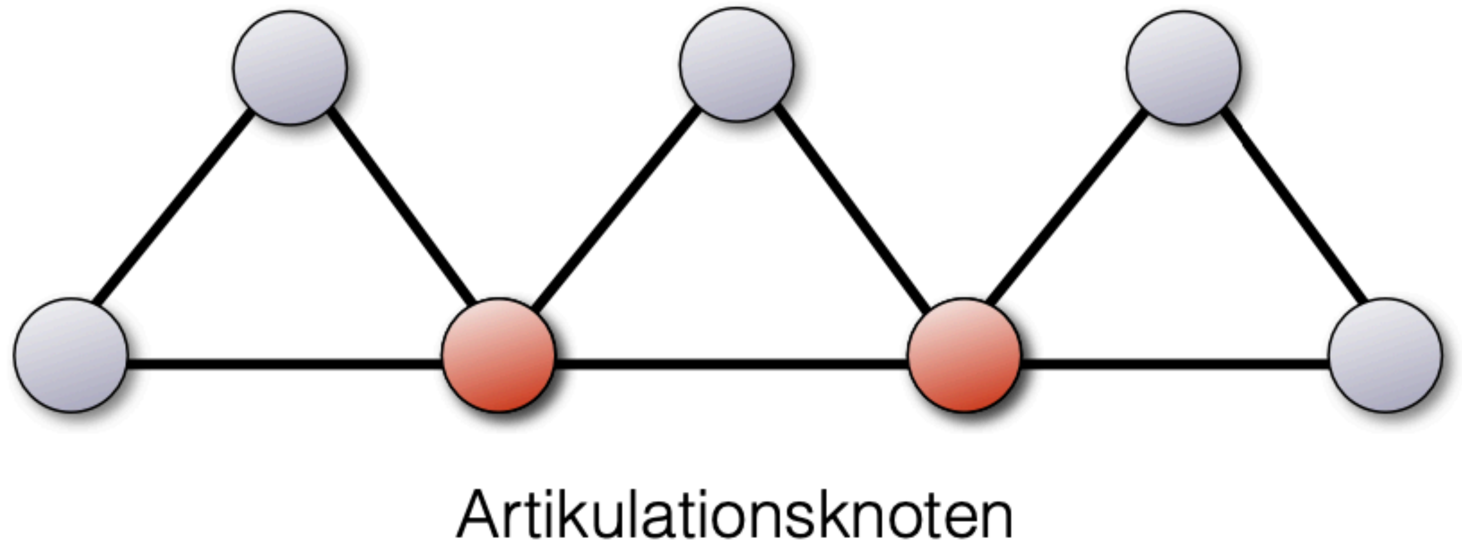
$\forall u, v \in V : \{u, v\}$ ist eine **Brücke** $\implies$

$\deg(u) = 1$ oder u ist ein **AK**

und

$\deg(v) = 1$ oder v ist ein **AK**

Umkehrung gilt nicht!

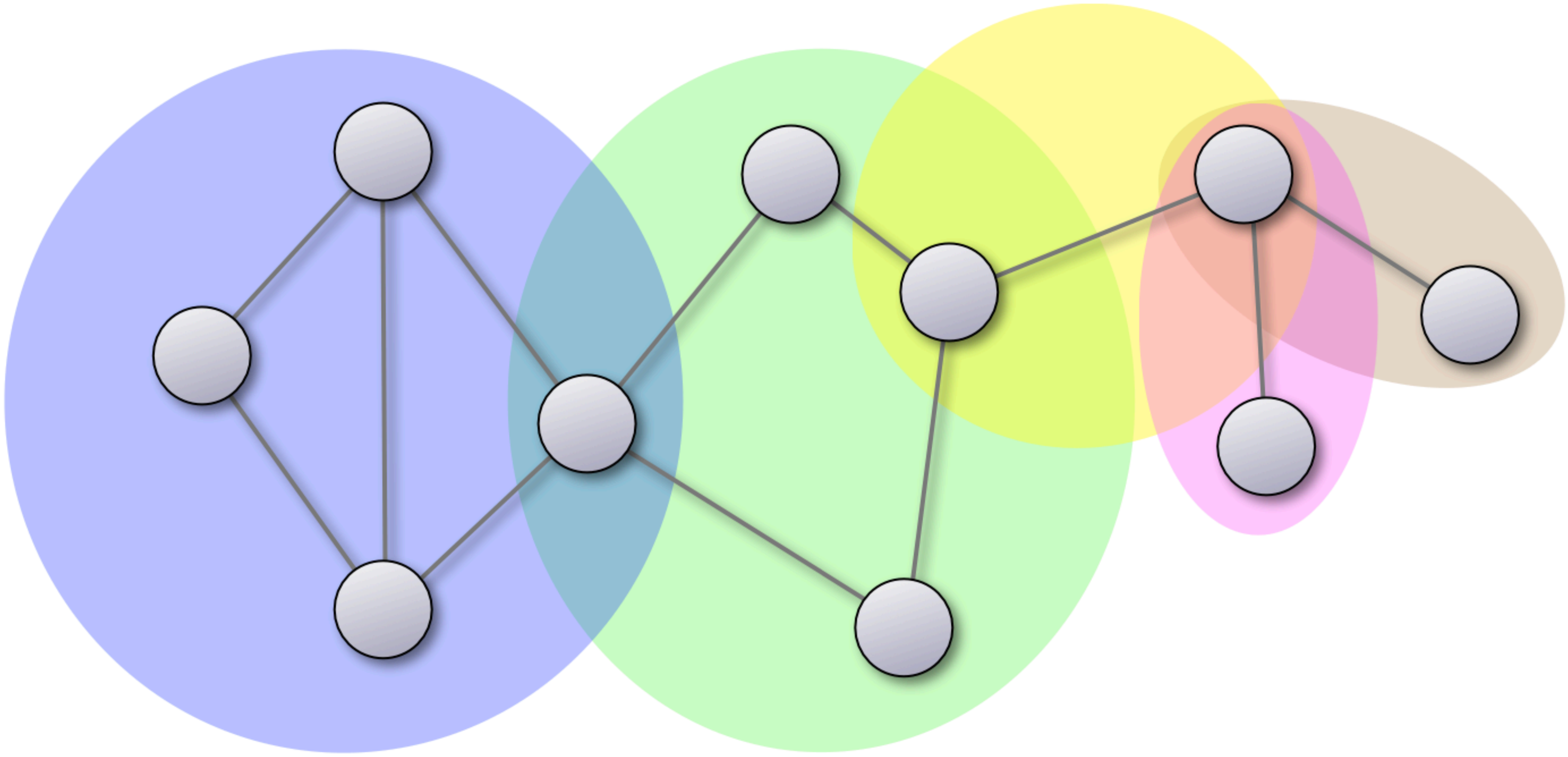


Blöcke

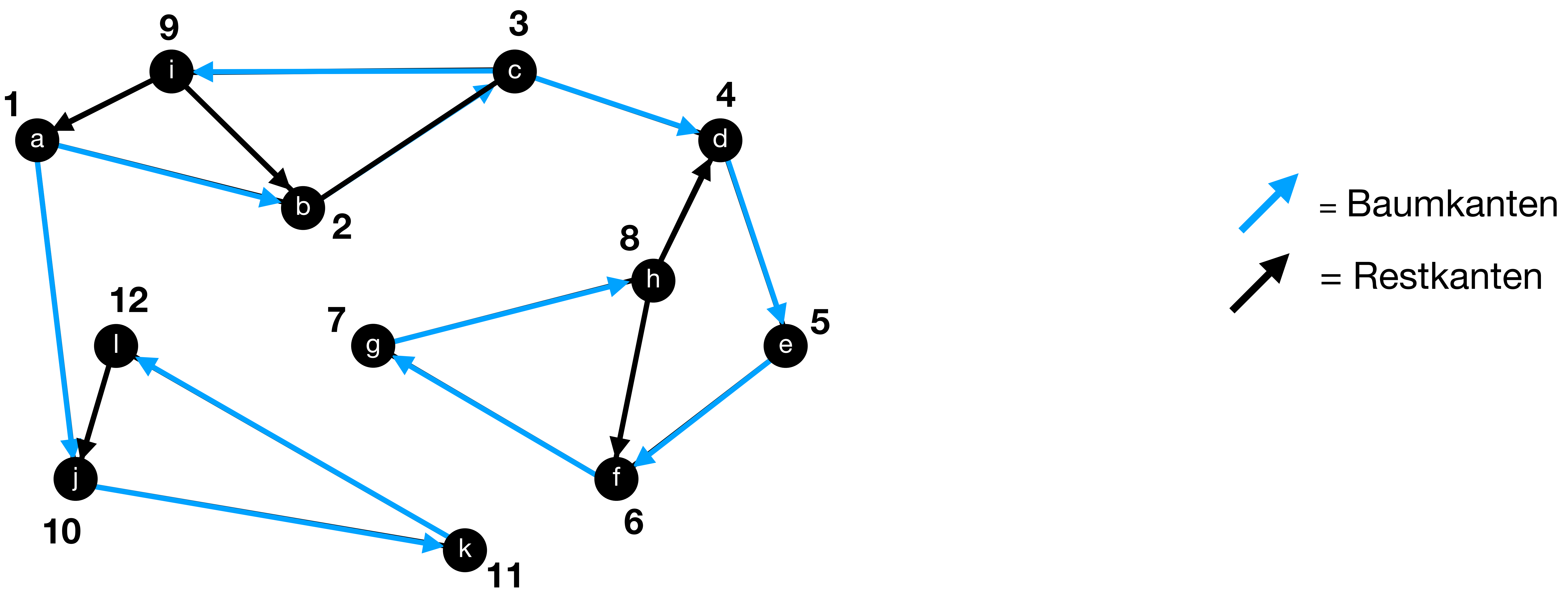
<u>Äquivalenzrelation</u> auf **Kanten**

$$e \sim f \iff \begin{cases} e = f \text{ oder} \\ \exists \text{ Kreis durch } e \text{ und } f \end{cases}$$

Lemma: Zwei Blöcke scheiden sich in einem **AK**



Depth First Search



Artikulationsknoten / Brücken finden

low[v]: kleinste **dfs**-Nummer, die man von v aus durch einen gerichteten Pfad aus beliebig vielen Baumkanten und maximal einer Restkante erreichen kann

→ Berechenbar in $O(|V| + |E|)$ mithilfe DP

$$\textbf{low}[v] = \min\left(\text{dfs}[v],\ \min_{(v,w)\in E} \begin{cases} \text{dfs}[w], & \text{falls } (v,w) \text{ Restkante} \\ \text{low}[w], & \text{falls } (v,w) \text{ Baumkante} \end{cases}\right)$$

AK finden

If 1) $v =$ **root** und v hat mind. 2 Kinder in DFS

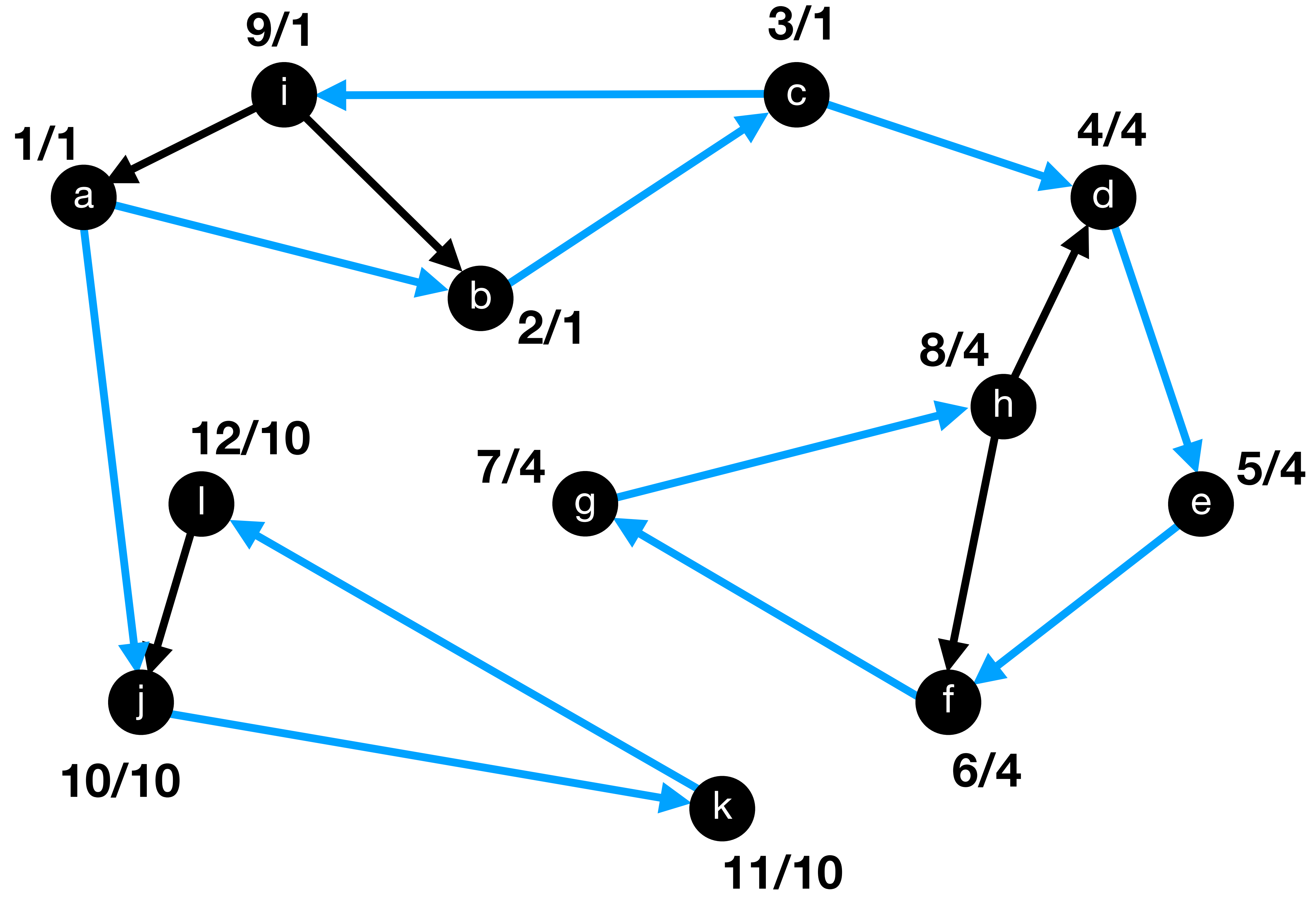
2) $v \neq$ **root** und v hat ein Kind w in DFS s.t. **low[w]** $\geq$ **dfs[v]**

Brücke finden

Eine Baumkante $e = (v, w)$ ist eine Brücke

$\iff$ **low[w]** $>$ **dfs[v]**

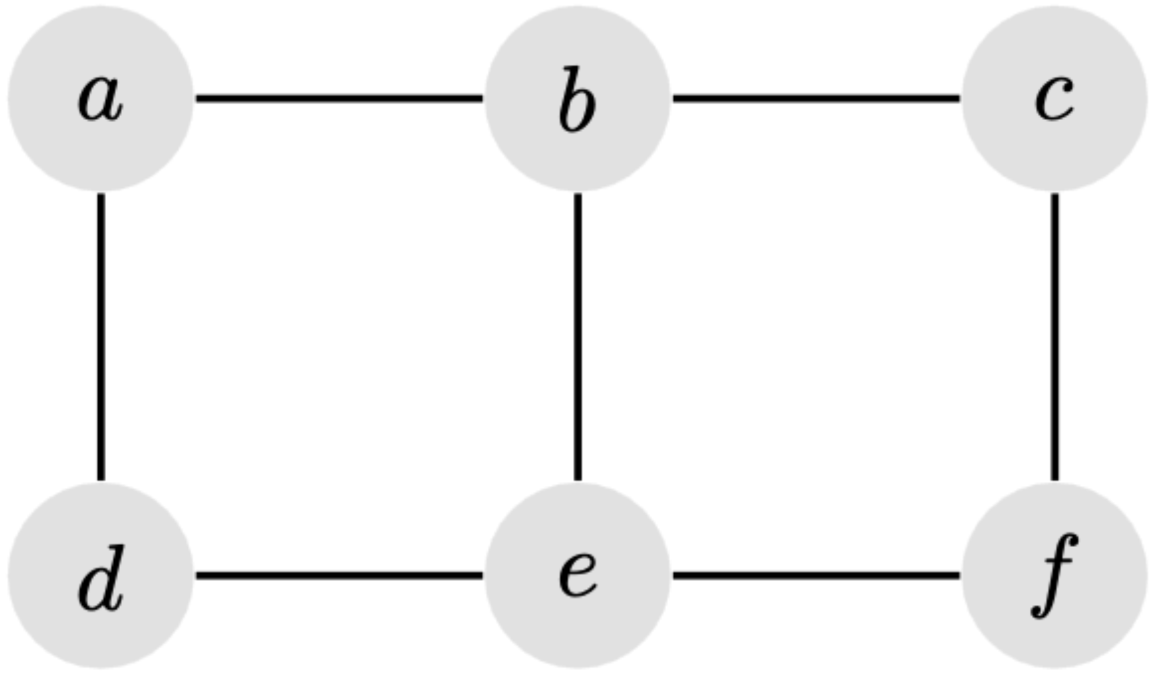
Eine Restkante ist nie eine Brücke



Aufgaben

Aufgabe 1 – *Pfade, Wege, Kreise*

Betrachten Sie folgenden Graphen $G = (V, E)$.

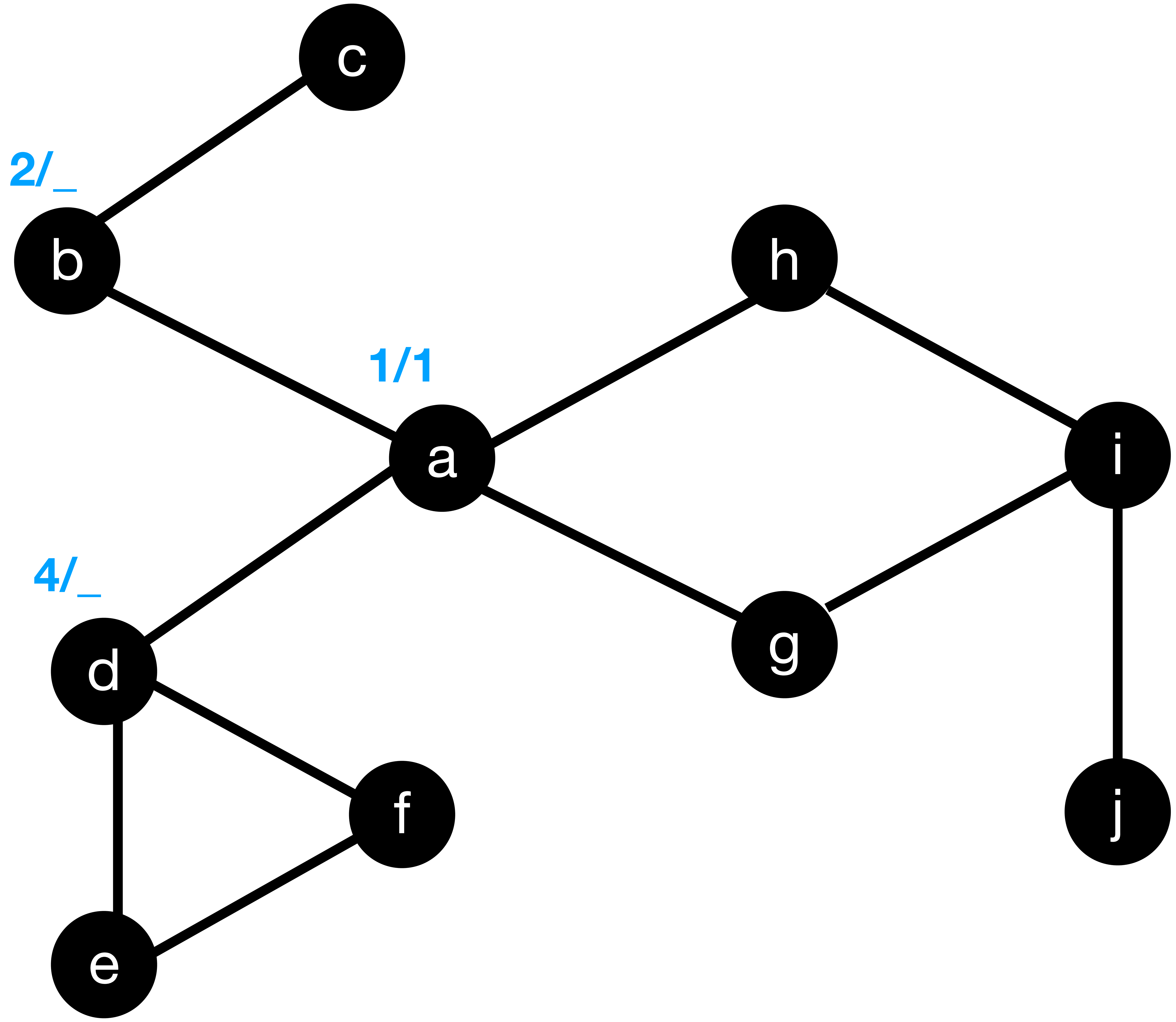


1. Welche Pfade der Länge 4 (d.h. mit 4 Kanten) gibt es von a nach e?
2. Welche Wege der Länge 4 (d.h. mit 4 Kanten) gibt es von a nach e?
3. Welche Kreise gibt es in G?
4. Wie viele Zykeln gibt es in G?

Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

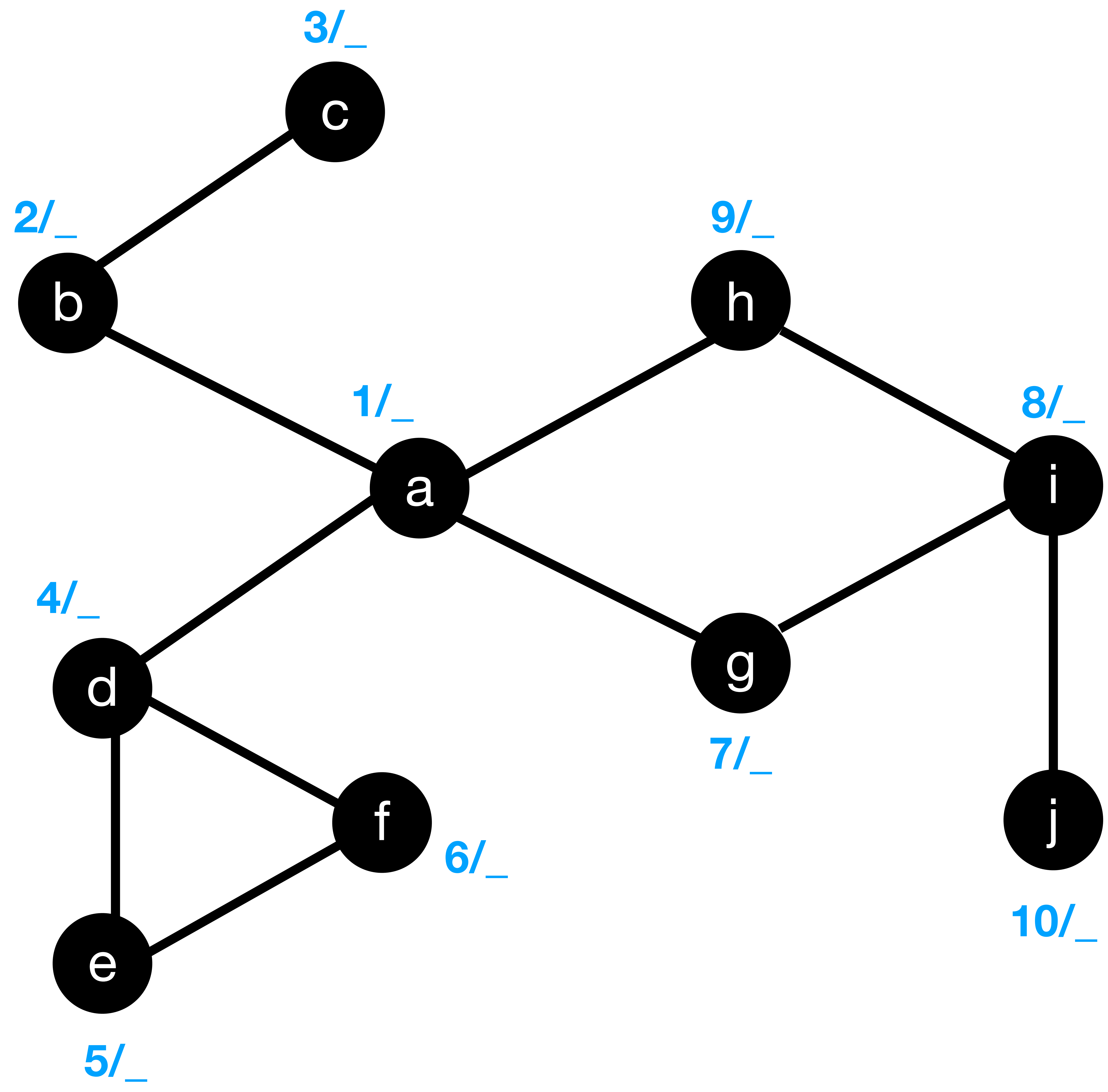
b) Welche Blöcke gibt es in diesem Graphen?



Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

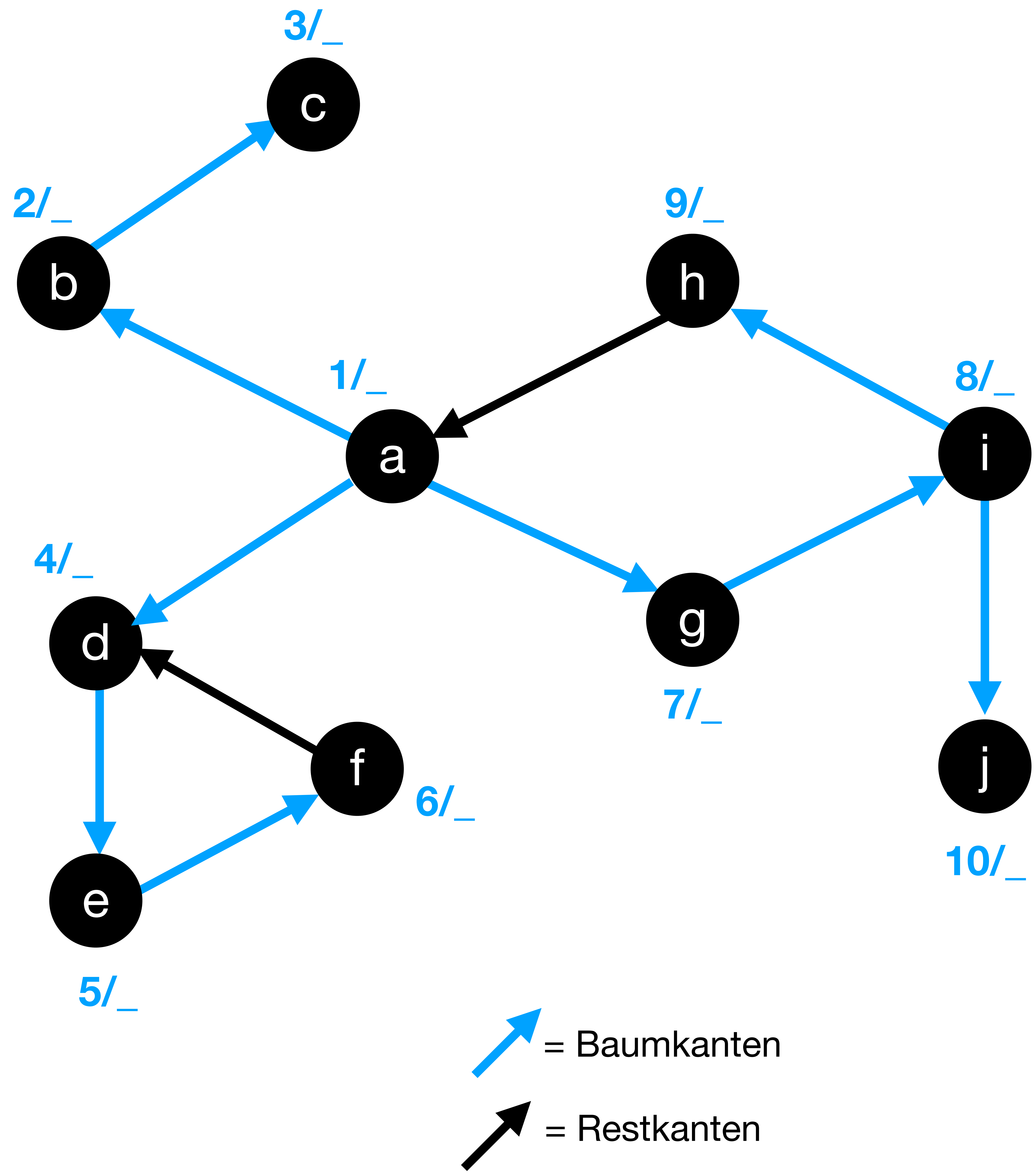
b) Welche Blöcke gibt es in diesem Graphen?



Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

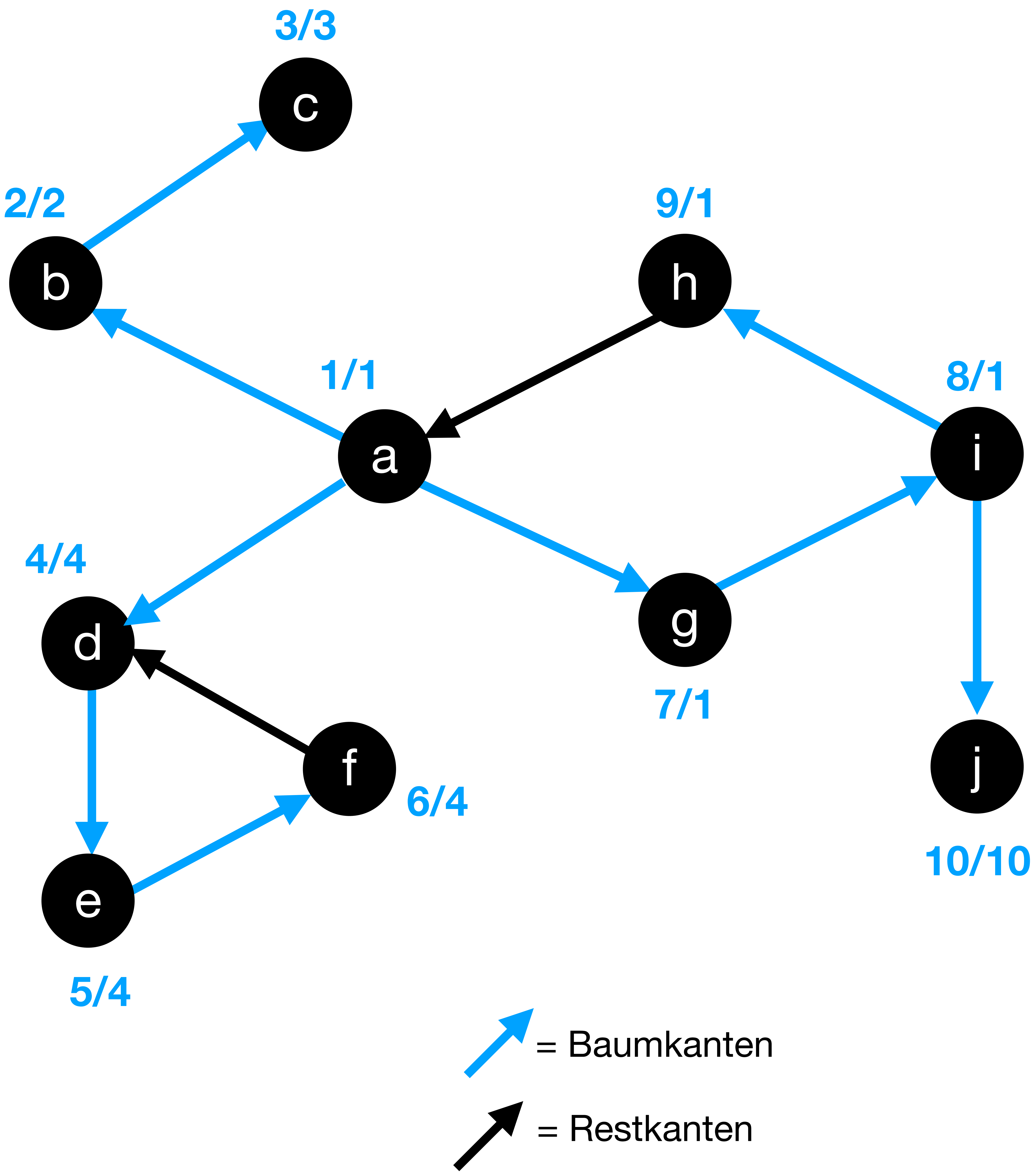
b) Welche Blöcke gibt es in diesem Graphen?



Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

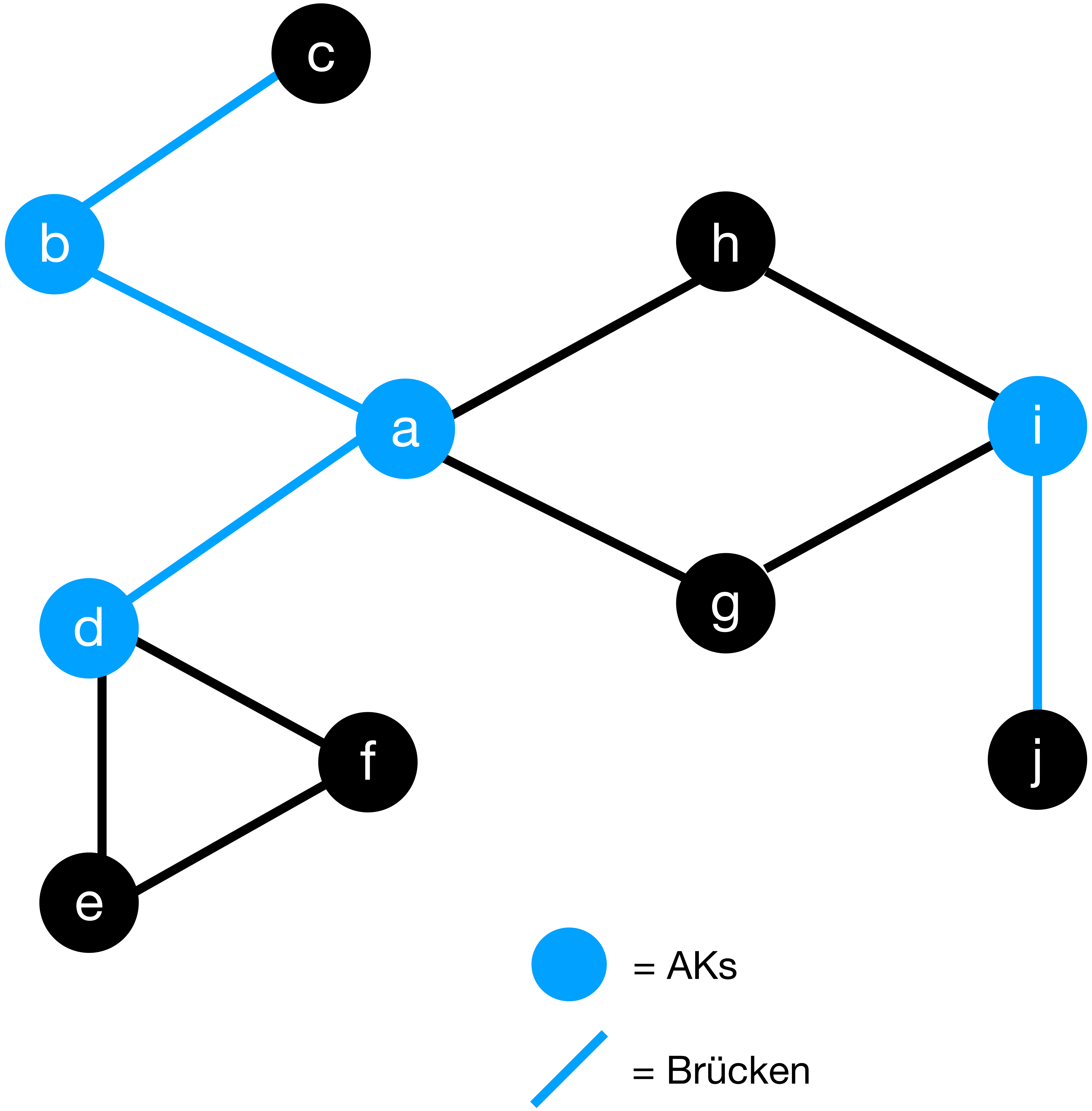
b) Welche Blöcke gibt es in diesem Graphen?



Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

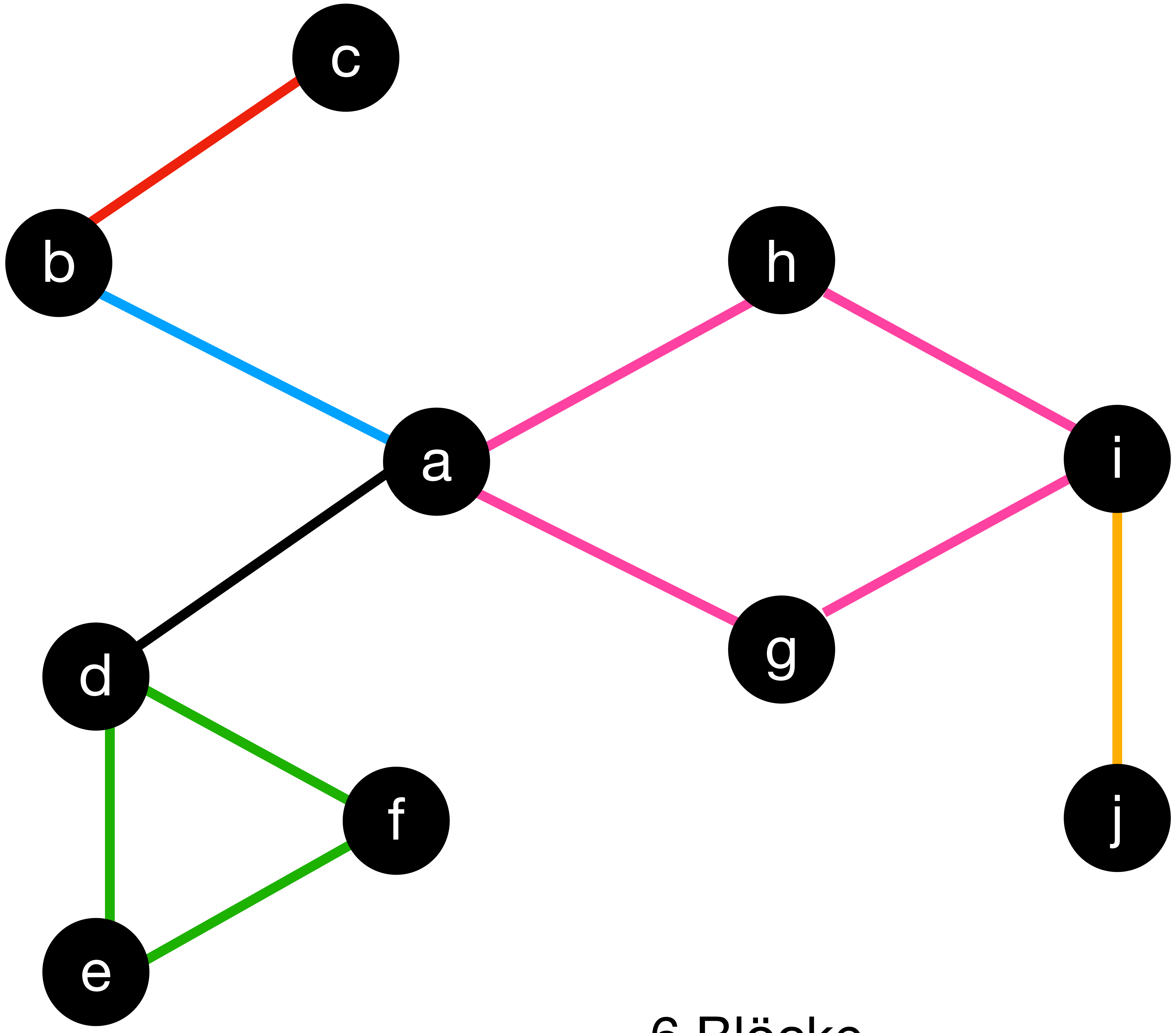
b) Welche Blöcke gibt es in diesem Graphen?



Aufgabe 2

a) Berechne die low/dfs Nummern ausgehend vom Knoten a und finde alle AKs/Brücken:

b) Welche Blöcke gibt es in diesem Graphen?



6 Blöcke

Aufgabe 3

Sei G ein Graph, der nur einen Block hat. Sind die folgenden Aussagen wahr/falsch?

i) G hat keine AKs

ii) G ist 2-zusammenhängend

iii) G ist 2-Kanten-zusammenhängend

iv) G kann Brücken haben

Aufgabe 3

Sei G ein Graph, der nur einen Block hat. Sind die folgenden Aussagen wahr/falsch?

i) G hat keine AKs

wahr -> ohne AK gibt es 2 ZHK,

somit kann es nicht einen Kreis durch alle Knoten gegeben haben

iii) G ist 2-zusammenhängend

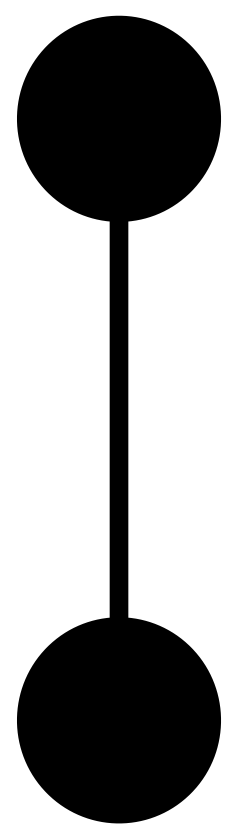
falsch ->

iv) G ist 2-Kanten-zusammenhängend

falsch ->

v) G kann Brücken haben

wahr ->



Aufgabe 4 – *Eine generelle Eigenschaft von Graphen*

Zeigen Sie, dass jeder Graph G mit $n \geq 2$ Knoten zwei Knoten $v \neq w$ enthält, sodass $deg(v) = deg(w)$.

Hinweis: Für ein gegebenes n, was ist der grösstmögliche Grad den ein Knoten haben kann?

Aufgabe 5 – *Algorithmus*

Beschreiben Sie einen Algorithmus der das folgende Problem löst: Gegeben ist die Eingabe bestehend aus einen Graphen $G = (V, E)$ mit n Knoten (gehen Sie davon aus, dass der Graph als Adjazenzliste gegeben ist). Ihr Algorithmus soll “Ja” ausgeben, falls G ein Baum ist und “Nein” andernfalls.

Wie immer wenn Sie einen Algorithmus beschreiben gehöhrt zu einer vollständigen Lösung: eine klare Beschreibung des Algorithmus, ein Korrektheitsbeweis und eine Laufzeitanalyse.

Hinweis: Für diese Aufgabe dürfen Sie das Statement aus Aufgabe 6 ohne Beweis verwenden.

Aufgabe 6 – *Charakterisierung von Bäumen (Challenge-Aufgabe)*

Zeigen Sie: Ist $G = (V, E)$ ein Graph auf $|V| \geq 1$ Knoten, so sind die folgenden Aussagen äquivalent:

(a) G ist zusammenhängend und kreisfrei (d.h. G ist ein Baum).

(b) G ist zusammenhängend und $|E| = |V| - 1$.

(c) G ist kreisfrei und $|E| = |V| - 1$.

(d) Für alle $x, y \in V$ gilt: G enthält genau einen x-y-Pfad.

Aufgabe 6 – *Charakterisierung von Bäumen (Challenge-Aufgabe)*

Zeigen Sie: Ist $G = (V, E)$ ein Graph auf $|V| \geq 1$ Knoten, so sind die folgenden Aussagen äquivalent:

(a) G ist zusammenhängend und kreisfrei (d.h. G ist ein Baum).

(b) G ist zusammenhängend und $|E| = |V| - 1$.

(c) G ist kreisfrei und $|E| = |V| - 1$.

(d) Für alle $x, y \in V$ gilt: G enthält genau einen x-y-Pfad.