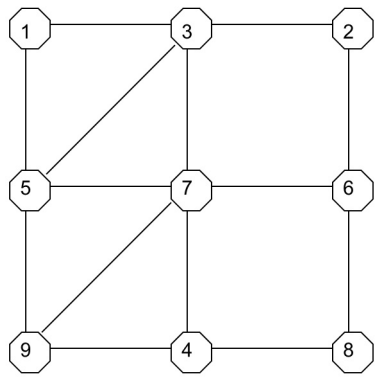
Graphen: Grundlegende Definitionen

- Graph G: abstrakte Struktur, die eine Menge von Objekten, oder *Knoten* V, mit zwischen diesen Objekten bestehenden Verbindungen, oder *Kanten* E, repräsentiert.
- *Notation:* Graph $G = (V, E)$ hat n Knoten und m Kanten.



Graphen: Grundlegende Definitionen

- *Mehrfachkanten* sind möglich (z.B. U-Bahn Schienennetz)
- *Gerichtete Graphen:* Jede Kante hat eine Richtung

  $u \longrightarrow v$ oder $[u, v\rangle$
- *Ungerichtete Graphen:* Kanten verlaufen in beide Richtungen

  $u \longleftrightarrow v$ oder $u \text{——} v$ oder $\{u, v\}$

  Kante in ungerichtetem Kanten entspricht Paar von ungerichteten Kanten

Graphen: Grundlegende Definitionen

- *Ungerichteter Graph* (V, E)*:* Bei Kante $e = \{u, v\} \in E$ sagt man u und v sind *Nachbarn*, oder u und v sind *adjazent*. Weiters u (und v) sind *inzident* zu e.
  $\mathrm{Neigh}(u)$ ist der Menge der Nachbarn von u. Der *Grad* von u ist $\deg(u) = |\mathrm{Neigh}(u)|$.
- *Gerichteter Graph* (V, E)*:*
  $\mathrm{Out}(u) = \{v \in V \mid [u, v\rangle \in E\}$ und $\mathrm{outdeg}(u) = |\mathrm{Out}(u)|$.
  $\mathrm{In}(u) = \{x \in V \mid [x, u\rangle \in E\}$ und $\mathrm{indeg}(u) = |\mathrm{In}(u)|$.

Repräsentation von Graphen

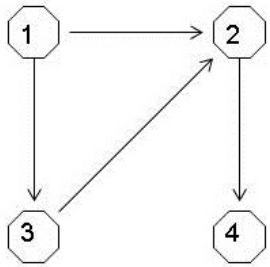
- *Adjazenzliste:* jeder Knoten u kennt eine Liste mit den Endpunkten der von u ausgehenden Kanten (also $\mathrm{Out}(u)$).
  - Platzbedarf $O(n+m)$
  - Um Frage zu beantworten, ob Kante $[u,v\rangle$ existiert, benötigt man $O(\mathrm{outdeg}(u)) = O(n)$ Zeit.
- *Adjazenzmatrix:* binäre Matrix der Dimension $n \times n$, die an Stelle $[u,v\rangle$ genau dann eine 1 hat, wenn die Kante (u,v) in G ist
  - Platzbedarf $O(n^2)$
  - Um Frage zu beantworten, ob Kante (u,v) existiert, benötigt man $O(1)$ Zeit
  - Bei ungerichteten ist die Adjazenzmatrix symmetrisch (und hat reelle Eigenwerte).

Graphen: Grundlegende Definitionen

- *Pfad:* Folge von Knoten $v_1, v_2, \ldots, v_k$, bei der aufeinanderfolgende Knoten v_i und v_{i+1} durch Kanten verbunden sind.
- *Weg:* Pfad, bei dem sich alle Knoten voneinander unterscheiden.
- *Zyklus:* Pfad, bei dem sich alle Knoten voneinander unterscheiden, außer daß erster und letzter Knoten übereinstimmen.
- Graph mit Zyklus heißt *zyklisch*. Graph ohne Zyklen heißt *azyklisch*.

Gerichtete Azyklische Graphen
Directed Acyclic Graphs (DAG)

- Kommen in vielen Anwendungen (z.B. Studienordnung) vor



- *Quelle:* Knoten mit Eingangsgrad Null
- *Senke:* Knoten mit Ausgangsgrad Null
- **Lemma:** Jeder DAG hat mindestens eine Quelle und mindestens eine Senke.

  Beweis: Betrachte Pfad mit maximaler Anzahl von Kanten: kann keinen Zyklus enthalten; erster Knoten auf Pfad muss Quelle sein, letzter eine Senke.

Topologische Sortierung

- Knotenordnung, in der für jede gerichtete Kante $[u, v\rangle$ der Knoten u vor dem Knoten v in der Ordnung kommt, heißt *topologische Sortierung*
- **Satz:** Ein gerichteter Graph G besitzt genau dann eine topologische Sortierung, wenn G azyklisch ist.
- Algorithmus zum Finden einer topologischen Sortierung (Laufzeit $O(n + m)$):
  1. Finde eine Quelle q von G
  2. Füge q als nächstes Element in der Sortierung ein
  3. Entferne q und alle von q ausgehenden Kanten von G
  4. Wiederhole bis G keine Knoten mehr enthält

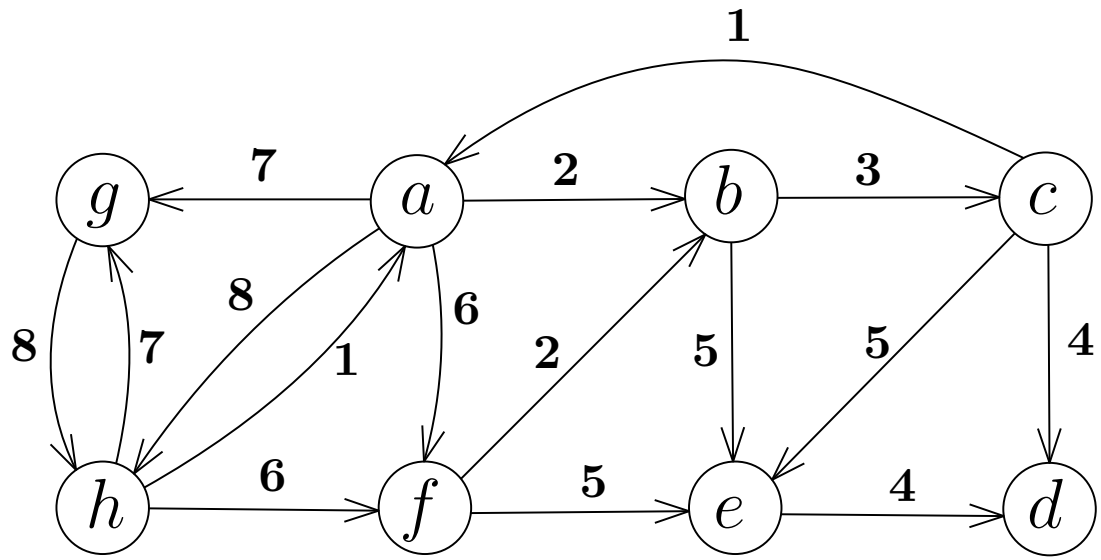
  Laufzeit $O(n + m)$, wenn man $\mathrm{indeg}(v)$ für jeden Knoten v aufrechterhält sowie die Menge aller Quellen.

~~Tiefensuche oder Depth First Search (DFS)~~
Lexikographisch kleinste Wege

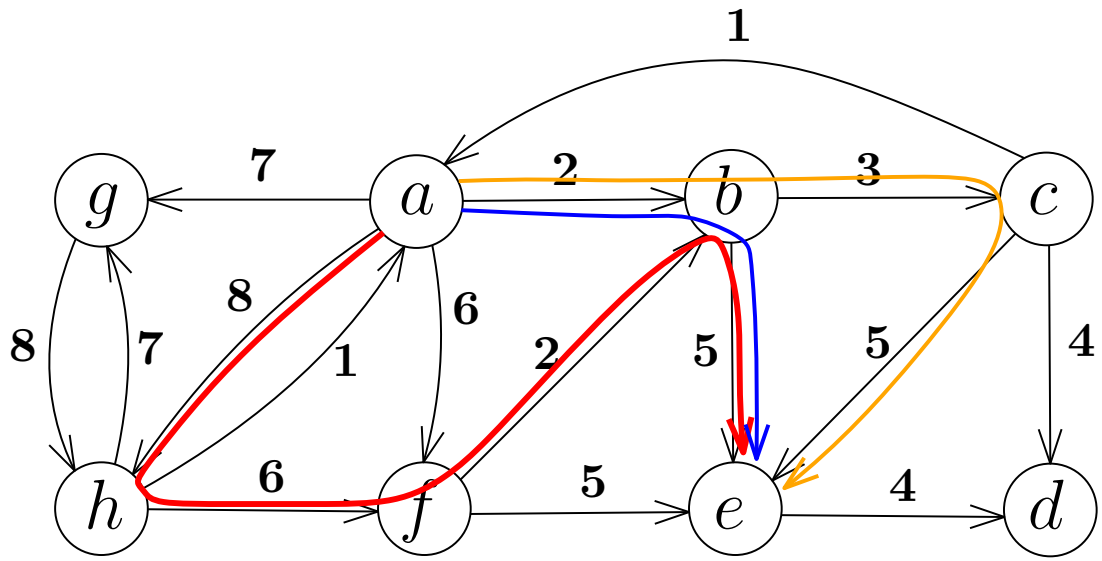
- Lexikographische Vergleiche: L geordnete Menge
  Strings $a = a_1 a_2 \cdots a_k$ und $b = b_1 b_2 \cdots b_\ell$ aus L^*
  $a \prec b$ (a lexikographisch kleiner als b) wenn
  1. $k = 0$ und $\ell > 0$ oder sonst
  2. $a_1 < b_1$ oder sonst
  3. $a_1 = b_1$ und $a_2 \cdots a_k \prec b_2 \cdots b_\ell$
- $\mu : E \longrightarrow L$ Kantenbeschriftung in gerichtetem Graphen $G = (V, E)$ ist *valide* wenn $\mu([u, v\rangle) = \mu([u, w\rangle) \Longrightarrow v = w$
  also, verschiedene v verlassende Kanten haben verschiedene Beschriftungen
- Pfadbeschriftung: p sei Pfad $v = v_0, v1, \cdots, v_k = w$ von v nach w
  $\mu(p)$ ist der String $\mu([v_0, v_1\rangle)\mu([v_1, v_2\rangle) \cdots \mu([v_{k-1}, v_k\rangle)$
- Pfadvergleich: p und q zwei Pfade die gemeinsamen Anfange habe. p ist kleiner als q, wenn $\mu(p) \prec \mu(q)$.

Lexikographisch kleinste Wege

Graph mit valider Kantenbeschriftung (durch Knotenordnung induziert)



Lexikographisch kleinste Wege



Drei Pfade von a nach e:

$p = [a, h, f, b, e\rangle$ mit Beschriftung 8,6,2,5

$q = [a, b, e\rangle$ mit Beschriftung 2,5

$r = [a, b, c, e\rangle$ mit Beschriftung 2,3,5

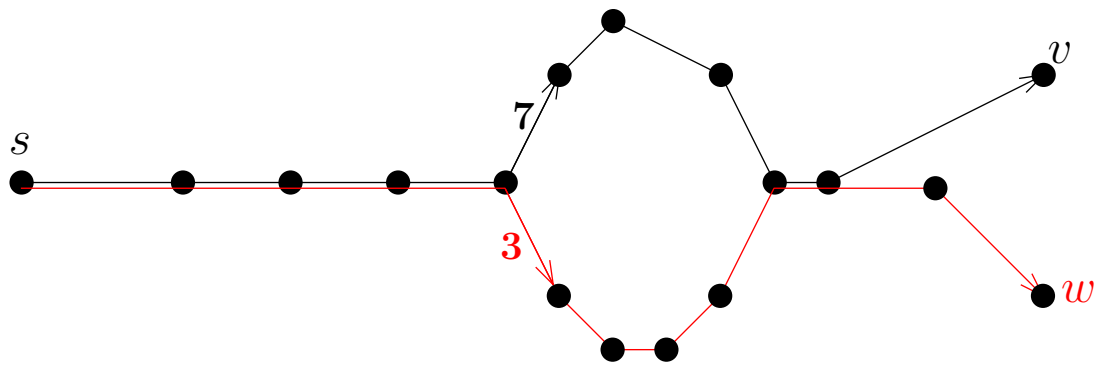
r ist der lexikographisch kleinste dieser drei Pfade, und auch insgesamt der kleinste *aller* Pfade von a nach e

Lexikographisch kleinste Wege

Notation: $\mathrm{lkW}_s(v)$ sei der lex-kleinste-Weg von s nach v.

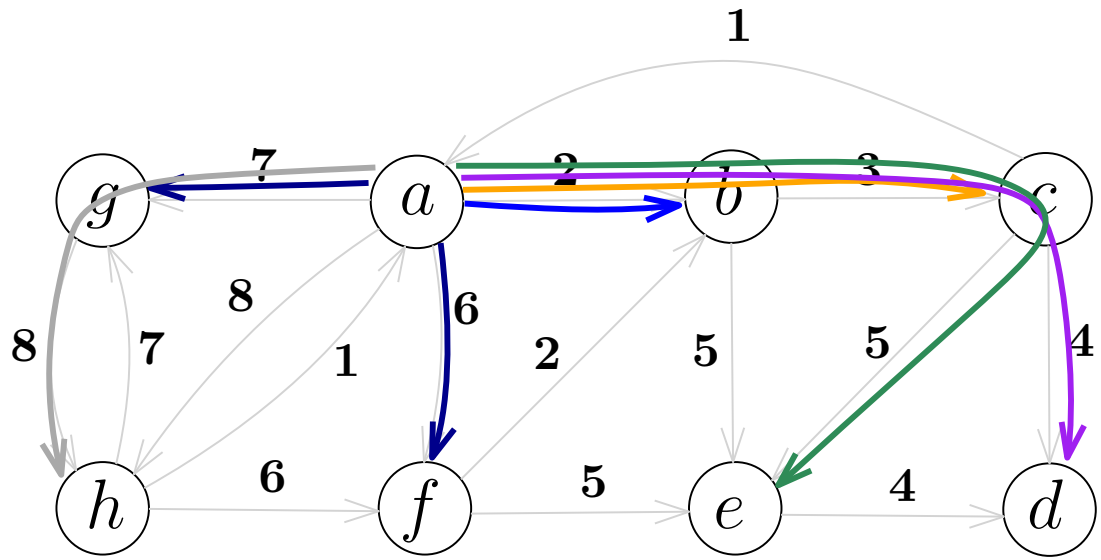
Präfix-Lemma: Die Knoten, die sowohl auf $\mathrm{lkW}_s(v)$ wie auch auf $\mathrm{lkW}_s(w)$ liegen, bilden einen gemeinsamen Präfix (Anfangsstück) dieser beiden Wege.

Beweis: Anfangsknoten s auf beiden Wegen. Wenn die gemeinsamen Knoten keinen gemeinsamen Präfix bilden, dann ist mindestens einer der beiden Wege kein lex-kleinster Weg.



Lexikographisch kleinste Wege

Korrolar des Präfix-Lemmas: $G = (V, E)$ gerichteter Graph, $s \in V$ und $V_s \subset V$ die Menge der Knoten, die von s erreichbar sind (d.h. es gibt einen Weg)
Die Menge alles lex-kleinsten Wege, die in s beginnen, also $\{\mathrm{lkW}_s(v) \mid v \in V_s\}$ bilden bilden einen Baum mit Wurzel s, den *lex-kleinsten-Wege-Baum T_s von s.*
T_s ist ein aufspannender Baum von V_s.



Augmentierung eines gerichteten Graphens

$G = (V, E)$ gerichteter Graph
(Quellen)-Augmentierung von G zu $G_\Diamond = (V_\Diamond, E_\Diamond)$ mit
$V_\Diamond = V \cup \{\Diamond\}$ und $E_\Diamond = E \cup \{[\Diamond, v\rangle \mid v \in V\}$

In $G_\Diamond$ sind alle Knoten von $\Diamond$ erreichbar. Die lex-kleinste-Wege-Baum $T_\Diamond$ ist ein aufspannender Baum von $V_\Diamond$ und induziert einen aufspannenden Wald von V.

Nummerierungen von Knoten in Bäumen

T ein Baum, geordnet, in dem Sinn, dass für jeden Knoten die Kinder geordnet sind (z.B. über Kantenbeschriftung)

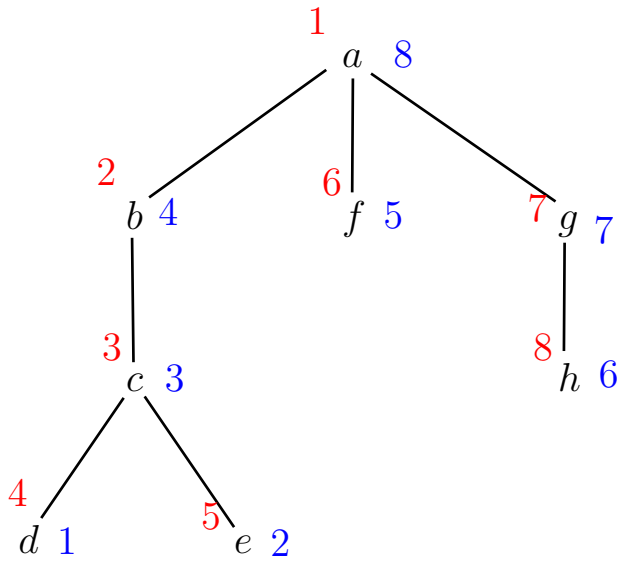
Präordnung: Nummeriere zuerst die Wurzel und dann rekursiv den Teilbaum jedes Kindes (entsprechend der Kinderordnung)

Postordnung: Nummeriere zuerst rekursiv die Kinder (entsprechend der Kinderordnung) und zuletzt die Wurzel

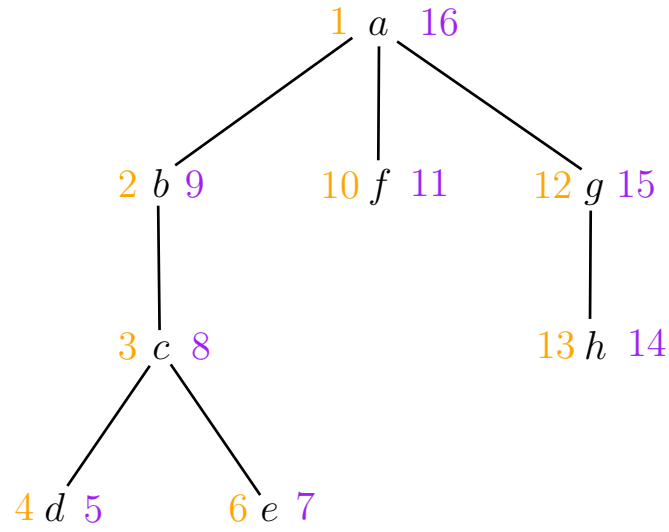
***d*-*f*-Nummerierung:**

```
global variable z:=0
dfnum(root of T) where
  dfnum(v) =
     d[v]:=++z
     foreach child w of v in order do dfnum(w) od
     f[v]:=++z
```

Nummerierungen von Knoten in Bäumen



Präordnungsnummerierur
Postordnungsnummerieru



d-f Nummerierung

Bei einem lex-kleinster-Wege-Baum T_s ist die Präordnungszahl jedes Knoten v genau der Rang von $\mathrm{lkW}_s(v)$ in der lexikographischen Ordnung aller lex-kleinsten Wege aus s, also $\{\mathrm{lkW}_s(v) \mid v \in V_s\}$.

Lexikographisch kleinste Wege

Nachkommen-Lemma: $G = (V, E)$ gerichteter Graph, $s \in V$ und T_s lex-kleinster-Wege-Baum

Es sei $v \in V$ und es sei $w \in V$ von v erreichbar und es gelte $\mathrm{lkW}_s(v) \prec \mathrm{lkW}_s(w)$.
Dann ist $\mathrm{lkW}_s(v)$ ein Präfix von $\mathrm{lkW}_s(w)$, oder anders gesagt, w ist ein Nachkomme von v im Baum T_s.

Beweis: Nimm an, die Aussage stimmt nicht, und $s = v_0, v_1, \ldots, v_k = v$ sei $\mathrm{lkW}_s(v)$ und $s = v_0, \cdots, v_h, w_{h+1}, \cdots, w_\ell = w$ sei $\mathrm{lkW}_s(w)$ mit $h < k$.
Wenn $\mu([v_h, w_{h+1}\rangle) < \mu([v_h, v_{h+1}\rangle)$, dann widerspricht das der Annahme $\mathrm{lkW}_s(v) \prec \mathrm{lkW}_s(w)$.

Wenn $\mu([v_h, w_{h+1}\rangle) > \mu([v_h, v_{h+1}\rangle)$, dann wäre der Weg von s nach w, der sich aus $\mathrm{lkW}_s(v)$ und dem wegen der angenommenen Erreichbarkeit existierenden Weges von v nach w ergibt, lexikographisch kleiner als $\mathrm{lkW}_s(w)$, also $\mathrm{lkW}_s(w)$ nicht richtig.