KIT
Institut für Theoretische Informatik

Jun.-Prof. Henning Meyerhenke, Jun.-Prof. Dennis Hofheinz
Christian Staudt, Christoph Striecks

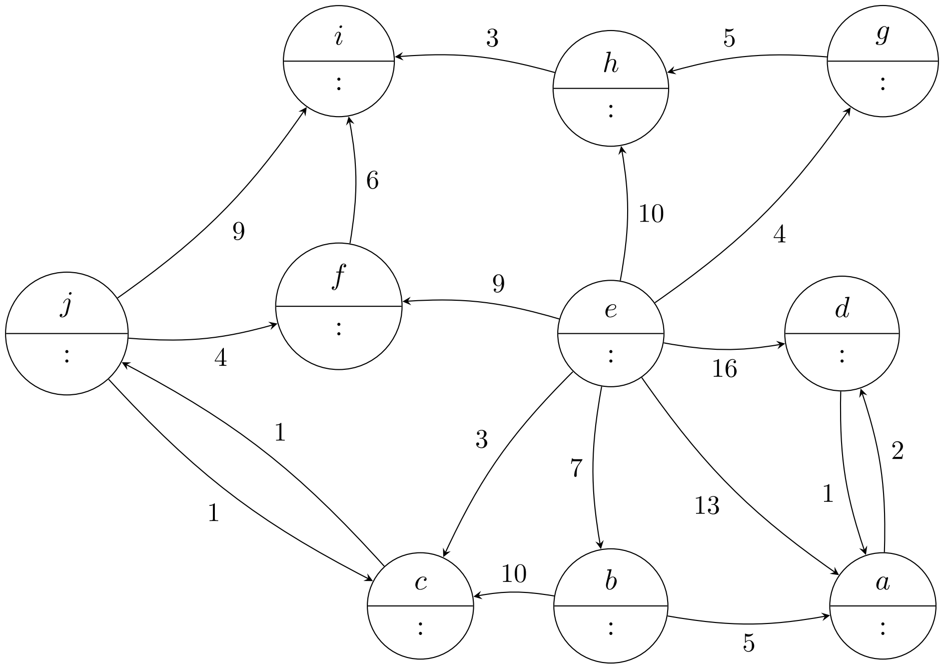
9. Übungsblatt zu Algorithmen I im SS 2015

https://crypto.iti.kit.edu/algo-sose15
{staudt,striecks}@kit.edu

Aufgabe 1 (*Dijkstras Algorithmus, 4 Punkte*)

Wir beschäftigen uns mit dem Algorithmus von Dijkstra aus der Vorlesung. Sei ein Graph $G = (V, E)$ wie unten dargestellt. Führen Sie Dijkstras Algorithmus auf G aus. Sie können die Darstellung von G benutzen, um die Ergebnisse nach der Ausführung darin einzutragen. Schreiben Sie dazu in jeden Knoten $v \in V$ rechts vom Doppelpunkt die Länge des kürzesten Pfads vom Startknoten e aus gesehen und links vom Doppelpunkt den Vorgänger auf diesem kürzesten Pfad. Zeichnen Sie zudem den Baum der kürzesten Wege von e zu allen anderen Knoten aus V in G ein.

Darstellung von G:



Aufgabe 2 (*Doktor Meta, 1 + 3 Punkte*)

Der ebenso geniale wie redselige Wissenschaftler und Superbösewicht Doktor Meta steht vor dem Triumph. Ihm ist es gelungen, den internationalen Spitzenagenten Sven van Hagen gefangen zu nehmen. Doktor Meta macht ihn sofort mit seiner neuen, unnötig komplizierten Laserapparatur bekannt. Er kann es sich nicht nehmen lassen, die Arbeitsweise der komplexen Maschine zu erklären. Er erläutert die Schritte, die notwendig sind, um die Maschine zu aktivieren, und wie diese aufeinander aufbauen. Spitzenagent van Hagen zweifelt an, dass Doktor Meta von ihm erwartet, komplizierte Abhängigkeiten aufzulösen. Sei

$$V = \{\text{Fusionsreaktor erhitzen}, \text{ Fluxkompensator aktivieren}, \text{ Laser ausrichten}, \text{ Plan erklären}, \\ \text{Linsenkappe entfernen}, \text{ Laser einschalten}\}$$

eine Menge von Arbeitsschritten. Sei

$$E = \{(\text{Linsenkappe entfernen}, \text{ Laser einschalten}), (\text{Laser ausrichten}, \text{ Laser einschalten}), \\ (\text{Fusionsreaktor erhitzen}, \text{ Laser einschalten}), (\text{Fluxkompensator aktivieren}, \text{ Laser einschalten})\}$$

eine Menge von Abhängikeiten. Zusammen kann (V, E) als gerichteter Graph G dargestellt werden, der mittels Adjazenzlisten realisiert ist. Wir sagen, dass eine Sequenz $S \in V^{|V|}$ für G gültig ist, falls

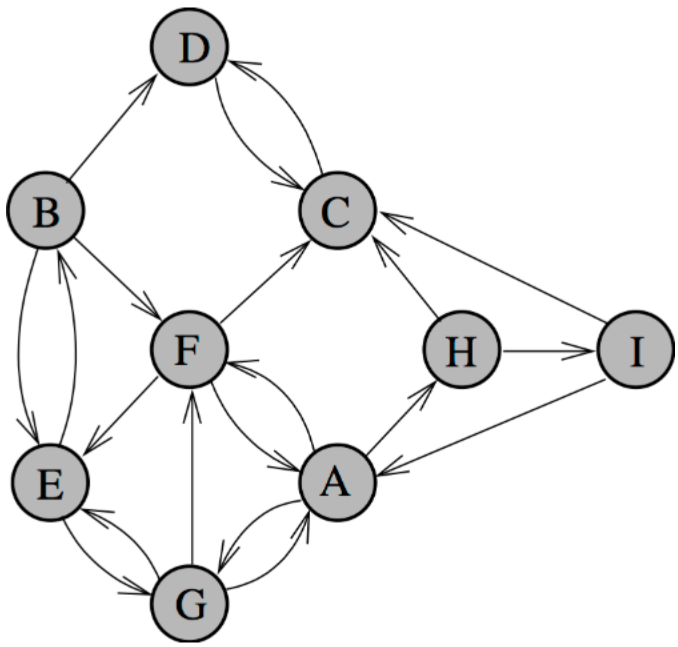
- jedes Element von V in S enthalten ist und
- für jede Kante $(x, y) \in E$ das Element x links von y in S steht. (Dabei sei $|V|$ die Kardinalität von V.)

a) Geben Sie eine gültige Sequenz S für G an.

b) Entwerfen Sie einen Algorithmus in Pseudocode, der für beliebige gerichtete Graphen $G' = (V', E')$ eine gültige Sequenz S' für G' ausgibt. Dabei sei S' als verkettete Liste realisiert. Gehen Sie vereinfachend davon aus, dass eine solche Implementierung einer verketteten Liste schon vorhanden ist und benutzt werden kann. (Hinweis: 3 Punkte kann es nur für einen Algorithmus mit Laufzeit $\mathcal{O}(|V| + |E|)$ geben.)

Aufgabe 3 (*Breitensuche,* $1 + 1 + 1 + 1$ *Punkte*)

Gegeben sei folgender gerichteter Graph mit der Knotenmenge $\{A, B, \ldots, I\}$:



In diesem Graph werde nun eine Breitensuche durchgeführt, und zwar ausgehend vom Knoten I.

a) Zählen Sie die Knoten des Graphen in einer Reihenfolge auf, in der diese von einer Breitensuche jeweils zum ersten Mal berührt werden. Heben Sie im Graph außerdem alle Kanten (z. B. farbig) hervor, die bezüglich dieser Breitensuche *tree*-Kanten sind.

b) Sind im Graph bzgl. dieser Breitensuche irgendwelche *backward*-Kanten vorhanden? Wenn ja, heben Sie diese hervor.

c) Sind im Graph bzgl. dieser Breitensuche irgendwelche *cross*-Kanten vorhanden? Wenn ja, heben Sie diese hervor.

d) Sind im Graph bzgl. dieser Breitensuche irgendwelche *forward*-Kanten vorhanden? Wenn ja, heben Sie diese hervor.

Ausgabe: Mittwoch, 17.6.2014
Abgabe: Freitag, 26.6.2014, 12:45 im Briefkasten im Untergeschoss von Gebäude 50.34