

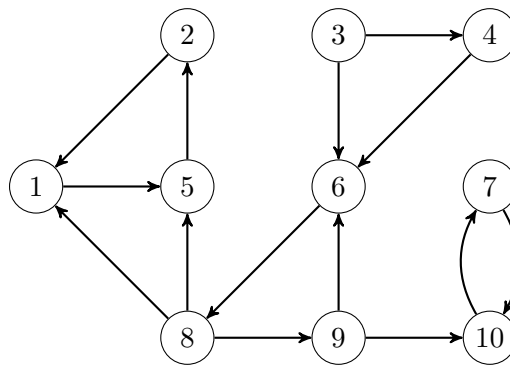
Aufgabe 10.1 (4 Punkte) Eine Liste der Knoten eines DAG $G = (V, E)$ heißt bzgl. \prec *topologisch sortiert*, wenn für alle $(u, v) \in E$ gilt, dass $u \prec v$. Es sei

$$u \prec_f v : \iff \text{finNum}[u] > \text{finNum}[v].$$

Beschreibe einen DFS-basierten Algorithmus, der zu einem gegebenen gerichteten Graphen G eine topologische Sortierung bzgl. \prec_f berechnet, wenn G ein DAG ist (s. Buch Abs. 9.2.1). Andernfalls sollte er einen Kreis ausgeben. Die Laufzeit soll dabei durch $O(|V| + |E|)$ beschränkt sein (mit Begründung).

Hinweis: Die Prozedur *traverseNonTreeEdge* kann zur Detektion des Kreises benutzt werden. Mache dich dazu die Kantentypen zunutze.

Aufgabe 10.2 (4 Punkte) Gegeben ist der folgende Graph G .



Führe den Algorithmus zum Auffinden der starken Zusammenhangskomponenten auf G aus. Gib den entstehenden vergrößerten Graphen G^S an. Benenne die Knoten von G^S entsprechenden der vom Algorithmus errechneten Repräsentanten.

Aufgabe 10.3 (4 Punkte) Sei ein gerichteter Graph $G = (V, E)$ gegeben. Beschreibe einen effizienten Algorithmus, der einen Graphen $G' = (V, E')$ berechnet mit folgenden Eigenschaften:

- a) G' und G haben die gleichen starken Zusammenhangskomponenten,
- b) G' und G haben den gleichen vergrößerten Graphen, und
- c) Die Anzahl von Kanten in G' ist möglichst klein.

Analysiere die Laufzeit von deinem Algorithmus.