

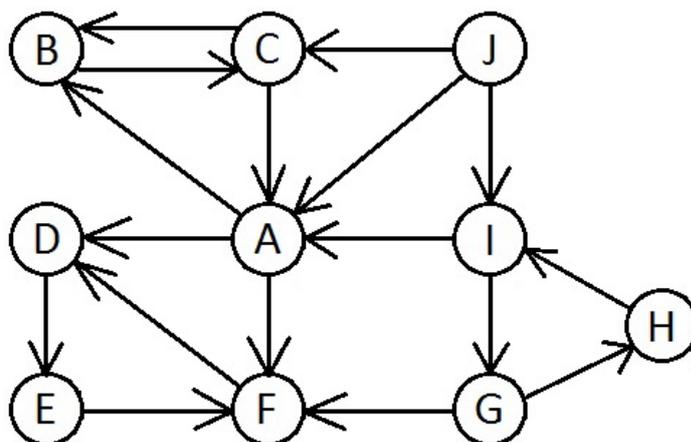
# Theoretische Informatik 1

## 6. Übung

Schreiben Sie Ihren Namen in den Dateinamen der von Ihnen abgegebenen Datei.

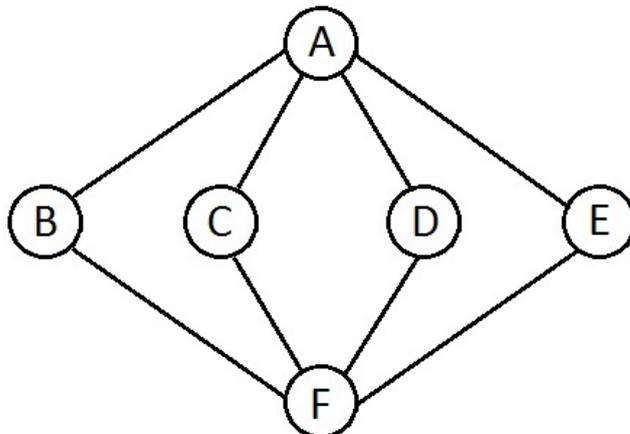
Abgabe: Lösen Sie die Aufgaben 1 und 2. Ihre Lösung senden Sie bitte bis zum Donnerstag, dem 23.11.23, 7:30 Uhr, per E-Mail an [knut.odermann@informatik.tu-chemnitz.de](mailto:knut.odermann@informatik.tu-chemnitz.de), am besten als pdf-Datei (idealerweise mit einem Textsatzsystem wie LaTeX erstellt) und gut lesbar. Nicht akzeptiert werden Scans bzw. Photos von Quellen in Formaten größer als DIN-A4. Vermerken Sie auf Ihrer Abgabe Ihren Namen, Ihre Matrikelnummer und Ihren Studiengang.

**1. Aufgabe** [5 Punkte] Finden Sie in folgendem gerichteten Graphen die starken Zusammenhangskomponenten mithilfe der Tiefensuche, Umkehrung der Orientierung der Kanten und anschließender erneuter Tiefensuche ab dem Knoten mit der größten Beendezeit (so, wie es in der Vorlesung vorgeführt wurde bzw. wie es im Skript auf S. 64 beschrieben ist).



Die Nachbarn eines Knoten sollen in lexikographisch aufsteigender Reihenfolge bearbeitet werden, d.h., beim Knoten A beginnend soll zuerst der Knoten B besucht werden, und nicht der Knoten D oder Knoten F, usw. (damit jeder denselben DFS-Baum erhält).

**2. Aufgabe** [5 Punkte] Führen Sie in folgendem ungerichteten Graphen, beginnend beim Knoten  $A$ , die Tiefensuche durch und geben Sie die Entdeckzeiten  $d(v)$ , die Beendezeiten  $f(v)$  sowie die Low-Werte der Knoten an.



**3. Aufgabe** Sei  $n$  eine natürliche Zahl.

Geben Sie einen 2-fach zusammenhängenden Graphen  $G = (V, E)$  an, so dass die Tiefensuche in  $G$ , unabhängig vom Startknoten und der Reihenfolgen in den Adjazenzlisten, immer einen Tiefensuchbaum ergibt, der einen Knoten mit mindestens  $n$  Kindern hat.