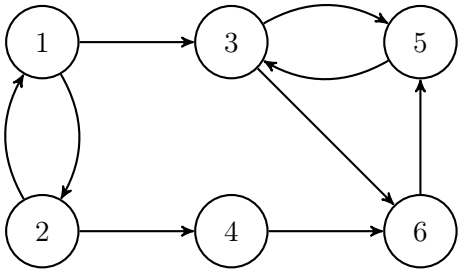
# Algorithmen und Datenstrukturen II
# WiSe 2026/27 – Serie 2

## 1 Tarjan-Algorithmus (6 Punkte)

Gegeben sei der Graph $G$:




Benutzen Sie den Tarjan-Algorithmus beginnend beim Knoten **1**, um die starken Zusammenhangskomponenten von $G$ zu berechnen. In der FOR-Schleife innerhalb von `Tarjan-visit` werden die Kindknoten $u$ des aktuellen Knotens $v$ in aufsteigender Reihenfolge (1,2,3,4,5,6) bearbeitet. Wenn also mehrere Möglichkeiten bestehen, so verwenden Sie den Knoten mit dem kleinsten Wert.

Geben Sie in der Reihenfolge der Abarbeitung des Algorithmus jeweils nach Beendigung der FOR-Schleife: `v`, `in`[v] und `l`[v] in Form der folgenden Tabelle an, wobei Sie bitte nicht benutzte Felder mit '-' befüllen.

| `v` | `in`[v] | `l`[v] |
|---|---|---|
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

Wie viele Starke Zusammenhangskomponenten hat der gegebene Graph?

## 2 Mengensysteme (3 Punkte)

Gegeben sind die Grundmenge $E = \{a, b, c, d\}$ und die folgenden Mengen von Mengen:

$$\mathcal{M}_1 = \{\, \emptyset,\, \{a\},\, \{b\},\, \{c\},\, \{d\},\, \{a,b\},\, \{a,c\},\, \{a,d\},\, \{b,c\},\, \{b,d\},\, \{c,d\},\, \{a,b,c\},\, \{a,b,d\},\, \{a,c,d\},\, \{b,c,d\},\, \{a,b,c,d\}\,\}$$

$$\mathcal{M}_2 = \{\, \emptyset,\, \{a\},\, \{b\},\, \{c\},\, \{d\},\, \{a,b\},\, \{a,c\},\, \{a,d\},\, \{b,c\},\, \{b,d\},\, \{c,d\},\, \{a,b,c\},\, \{a,b,d\},\, \{a,c,d\},\, \{a,b,c,d\}\,\}$$

$$\mathcal{M}_3 = \{\, \emptyset,\, \{a\},\, \{b\},\, \{c\},\, \{d\},\, \{b,c\},\, \{b,d\},\, \{c,d\},\, \{b,c,d\}\,\}$$

Geben Sie für jedes $i \in \{1, 2, 3\}$ an, ob $(E, \mathcal{M}_i)$ ein Mengensystem, ein Unabhängigkeitssystem oder ein Matroid ist. Fassen Sie Ihr Ergebnis in folgender Tabelle mit Einträgen ja/nein zusammen.

| | $\mathcal{M}_1$ | $\mathcal{M}_2$ | $\mathcal{M}_3$ |
|---|---|---|---|
| Mengensystem | | | |
| Unabh.System | | | |
| Matroid | | | |

## 3 Activity Selection (8 Punkte)

Bei einer Veranstaltung werden für die Besucher folgende Aktivitäten geplant:

| Aktivität | Startzeit | Endzeit |
|---|---|---|
| A | 7:00 Uhr | 8:00 Uhr |
| B | 1:00 Uhr | 3:00 Uhr |
| C | 1:00 Uhr | 6:00 Uhr |
| D | 2:00 Uhr | 4:00 Uhr |
| E | 4:00 Uhr | 8:00 Uhr |
| F | 5:00 Uhr | 7:00 Uhr |
| G | 4:00 Uhr | 5:00 Uhr |
| H | 6:00 Uhr | 9:00 Uhr |

Angenommen ein Teilnehmer möchte eine maximale Anzahl von Veranstaltungen besuchen, unabhängig deren Länge, und ohne Veranstaltungen verspätet zu erreichen oder verfrüht zu verlassen. Bestimmen Sie eine mögliche Menge von Veranstaltungen, die diesen Wunsch erfüllt mit Greedy Activity Selection. Geben Sie die Aktivitäten in der Reihenfolge an, in der sie betrachtet wurden. Sollte es mehrere Möglichkeiten zur Fortsetzung geben, betrachten Sie die Aktivitäten in alphabetisch aufsteigender Reihenfolge. Für nicht benötigte Felder verwenden Sie bitte "-".

## 4 Auftragsplanungsmatroid (3 Punkte)

Ermitteln Sie für das folgende Autragsproblem die optimale Lösung mittels des in den Vorlesungsfolien beschriebenen Kanonischen Greedy-Algorithmus.

Der Zeitbedarf für jeden Auftrag beträgt genau einen Tag. Die folgende Tabelle fasst die Aufträge, deren Gewinn und den Abgabetermin zusammen.

| Auftrag x | Gewinn w(x) | Termin d(x) |
|---|---|---|
| a | 4 | 3 |
| b | 6 | 1 |
| c | 10 | 2 |
| d | 3 | 3 |
| e | 8 | 2 |
| f | 7 | 1 |

Beschreiben Sie den Ablauf des Kanonischen Greedy-Algorithmus indem Sie die Aufträge in der Reihenfolge ihrer Bearbeitung angeben, und mit Ja oder Nein ob der jeweilige Auftrag der Lösungsmenge angefügt wird oder nicht.