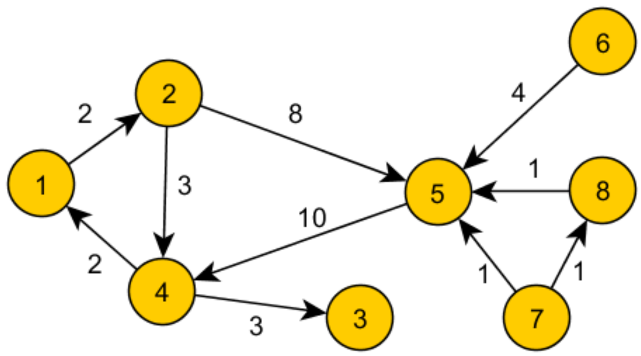
Arbeitsblatt: Graph Repräsentationen und top. Sortieren

Sie haben im Unterricht verschiedene Möglichkeiten zur Speicherung von Graphen kennengelernt. In diesem Arbeitsblatt sollen die verschiedenen Repräsentationen nochmals repetiert und im Anschluss analysiert werden. Das Ziel ist, dass sie schlussendlich beurteilen können, in welchen Anwendungsfällen sich welche Repräsentation gut eignet.

1. Graph-Repräsentation

Geben ist der folgende gewichtete Graph:



Geben Sie dazu die drei Repräsentationen an:

- Adjazenzmatrix

	1	2	3	4	5	6	7	8
1		2						
2				3	8			
3								
4	2		3					
5				10				
6					4			
7					1			1
8					1			

- Kantenliste

from:	1	2	2	4	4	5	6	7	7	8
to:	2	4	5	1	3	4	5	5	8	5
weight:	2	3	8	2	3	10	4	1	1	1

- Adjazenzliste

1	2	3	4	5	6	7	8
↓	↓		↓	↓	↓	↓	↓
2/2	4/3		1/2	4/10	5/4	5/1	5/1
	↓		↓			↓	
	5/8		3/3			8/1	

2. Komplexitätsanalyse der verschiedenen Graph-Repräsentation

Geben Sie für die gegebenen Operationen und Repräsentationen die Komplexitätsklassen an. Nun haben wir nicht wie bisher n als Eingabemenge, sondern n Knoten und m Kanten.

Operation	**Adjazenzmatrix**	**Kantenliste (Array)**	**Kantenliste (Liste)**	**Adjazenzliste**
Einfügen eines neuen Knotens	$O(n^2)$ (komplette Matrix kopieren)	O(1) (muss nichts tun, Aufwand =0, konstant)	O(1) (muss nichts tun, Aufwand =0, konstant)	Je nach Art der Speicherung: Mit HashMap: O(1) Mit Array: O(n) (wenn Array vergrössert warden muss)
Einfügen einer neuen Kante	O(1)	O(m) (Alle Kanten in grösseres Array kopieren)	O(1) (Kante wird zu Beginn der Liste eingefügt)	O(1) (Kante wird zu Beginn der Liste beim Startknoten eingefügt)
Ändern des Gewichts einer Kante	O(1)	O(m) (Kante muss gesucht werden)	O(m) (Kante muss gesucht werden)	O(m) (Kante muss gesucht werden)
Löschen eines Knotens	$O(n^2)$	O(m) (Alle betroffenen Kanten müssen gesucht werden)	O(m) (Alle betroffenen Kanten müssen gesucht werden)	O(m) (nach zum Knoten führenden Kanten suchen)
Löschen einer Kante	O(1)	O(m)	O(m)	O(m)
Prüfen ob Kante zwischen Knoten A und B existiert	O(1)	O(m)	O(m)	O(m)

Bemerkung zu Adjazenzlisten: Das Suchen nach allen von einem Knoten wegführenden Kanten wird in der Tabelle für die Worst Case Situation eines sternförmigen Graphen mit O(m) angegeben. Im Mittel ist jedoch mit O(m/n) zu rechnen.

Geben Sie mit **T** (True) und **F** (False) für die Repräsentationen an, ob die gegebenen Operationen möglich sind.

Einfügen einer parallelen Kante	F	T	T	T
Speichern mehrerer Kanten-Eigenschaften (z.B Gewicht und Kosten)	T[1]	T[1]	T[1]	T[1]

[1]: Dazu muss wie bei den anderen Graph-Repräsentationen ein Edge-Objekt erstellt werden und dies in die Matrix abgelegt anstatt ein einzelner Integer-Wert.

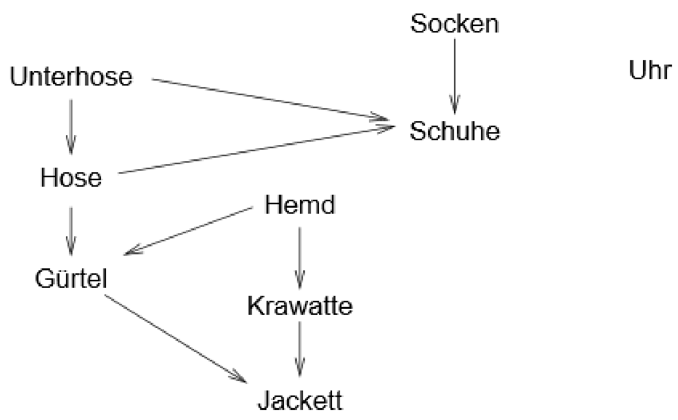
3. Topologisches Sortieren

a) Gibt es für den Graphen in der Aufgabe 1 eine topologische Sortierung? Begründen Sie.

Nein, der Graph enthält Zyklen und kann deshalb nicht topologisch sortiert werden.

b) Ein zerstreuter Professor hat am Morgen Probleme die Kleidung in der richtigen Reihenfolge anzuziehen. Daher notiert er die Reihenfolgebedingungen beim Anziehen:

- Unterhose vor Hose
- Hose vor Gürtel
- Hemd vor Gürtel
- Gürtel vor Jackett
- Hemd vor Krawatte
- Krawatte vor Jackett
- Socken vor Schuhen
- Unterhose vor Schuhen
- Hose vor Schuhen
- Uhr: Egal



Helfen Sie dem Professor!

Indegrees:

Item:	In-Degree
Uhr	0
Socken	0
Schuhe	~~3~~ ~~2~~ ~~1~~ 0
Unterhose	0
Hose	~~1~~ 0
Gürtel	~~2~~ ~~1~~ 0
Hemd	0
Krawatte	~~1~~ 0
Jackett	~~2~~ ~~1~~ 0

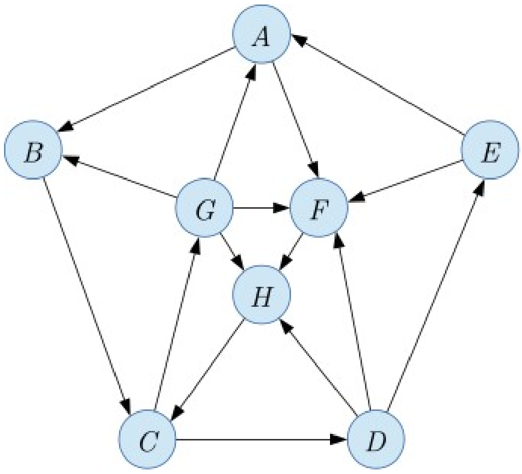
Queue: Uhr, Socken, Unterhose, Hemd, Hose, Krawatte, Schuhe, Gürtel, Jackett

Top. Sortierung: Uhr, Socken, Unterhose, Hemd, Hose, Krawatte, Schuhe, Gürtel, Jackett

4. Tiefensuche und Breitensuche

Die Reihenfolge, in der eine Tiefen- bzw. Breitensuche die Knoten des Graphs besucht, bezeichnen wir als *Tiefen*- bzw. *Breitenordnung*.

Für den untenstehenden Graphen nehmen wir an, dass er als Adjazenzlisten gespeichert ist, worin die ausgehenden Kanten eines Knotens in alphabetisch aufsteigender Reihenfolge vorkommen.



- Stellen Sie den obenstehenden Graphen als Adjazenzlisten dar.

A	B	C	D	E	F	G	H
↓	↓	↓	↓	↓	↓	↓	↓
B	C	D	E	A	H	A	C
↓		↓	↓	↓		↓	
F		G	F	F		B	
			↓			↓	
			H			F	
						↓	
						H	

- Geben Sie die Tiefen- und die Breitenordnung des Graphen, startend vom Knoten A, an.

Tiefenordnung: A, B, C, D, E, F, H, G
Breitenordnung: A, B, F, C, H, D, G, E

- Gibt es einen Startknoten in diesem Graphen, von dem aus die Tiefenordnung der Breitenordnung entspricht und umgekehrt? Begründen Sie Ihre Antwort.

Nein. Die Ordnungen wären dann gleich, wenn es in jedem Knoten jeweils nur eine Kante zu einem noch unbesuchten Knoten gäbe.

