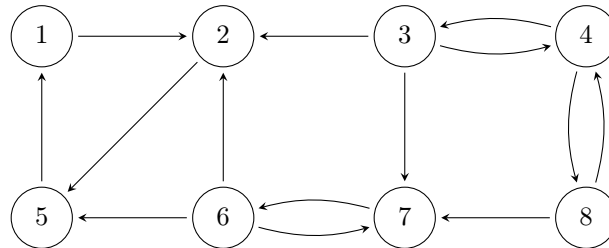


Tutoraufgabe 1 (SCC):

Wenden Sie Sharir's Algorithmus an (siehe Folien zur Vorlesung) um die starken Zusammenhangskomponenten des folgenden Graphen zu finden. Geben Sie das Array `color` und den Stack `S` nach jeder Schleifeniteration der ersten und zweiten Phase (also nach Zeile 17 und nach Zeile 22) an, falls DFS1 bzw. DFS2 ausgeführt wurde. Geben Sie zudem das Array `scc` nach jeder Schleifeniteration der zweiten Phase (also nach Zeile 22) an, falls DFS2 ausgeführt wurde. Nehmen Sie hierbei an, dass `scc` initial mit Nullen gefüllt ist und der Knoten mit Schlüssel i in der Adjazenzliste den $(i - 1)$ -ten Eintrag hat, also der Knoten mit Schlüssel 1 vom Algorithmus als erstes berücksichtigt wird usw.



Lösung: _____

Phase 1:

S: 5, 2, 1

color: (1, s), (2, s), (3, w), (4, w), (5, s), (6, w), (7, w), (8, w)

S: 5, 2, 1, 6, 7, 8, 4, 3

color: (1, s), (2, s), (3, s), (4, s), (5, s), (6, s), (7, s), (8, s)

Phase 2:

S: 5, 2, 1, 6, 7, 8, 4

color: (1, w), (2, w), (3, s), (4, s), (5, w), (6, w), (7, w), (8, s)

scc: (1, 0), (2, 0), (3, 3), (4, 3), (5, 0), (6, 0), (7, 0), (8, 3)

S: 5, 2, 1, 6

color: (1, w), (2, w), (3, s), (4, s), (5, w), (6, s), (7, s), (8, s)

scc: (1, 0), (2, 0), (3, 3), (4, 3), (5, 0), (6, 7), (7, 7), (8, 3)

S: 5, 2

color: (1, s), (2, s), (3, s), (4, s), (5, s), (6, s), (7, s), (8, s)

scc: (1, 1), (2, 1), (3, 3), (4, 3), (5, 1), (6, 7), (7, 7), (8, 3)

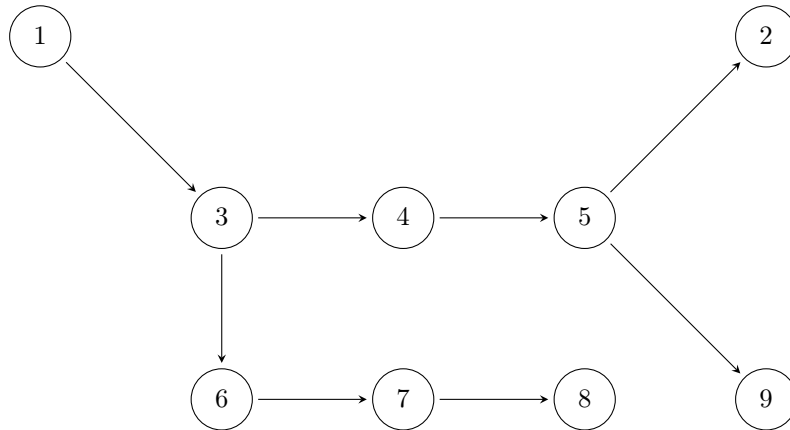
Der gegebene Graph hat folgende starken Zusammenhangskomponenten:

{1, 2, 5}
 {3, 4, 8}
 {6, 7}

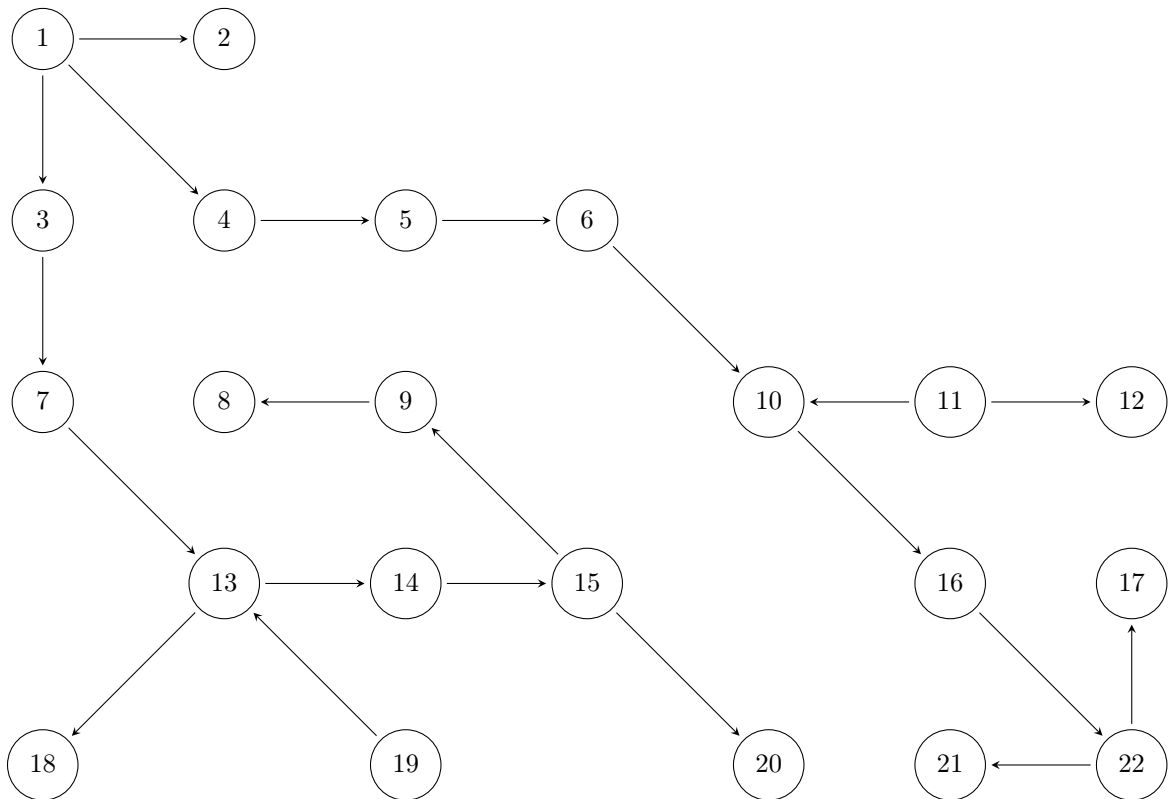
Tutoraufgabe 3 (Topologisches Sortieren):

Geben Sie eine topologische Sortierung des folgenden Graphen an. Dafür reicht es, eine geordnete Liste der Knoten mit dem dazugehörigen Topologiewert in Klammern anzugeben. Die Tiefensuche berücksichtigt bei mehreren Kindern diese in aufsteigender Reihenfolge (ihrer Schlüssel). Des Weiteren ist jedes Array, welches Knoten beinhaltet, aufsteigend nach deren Schlüsseln sortiert. Beachten Sie, dass der Knoten mit Schlüssel i in der Adjazenzliste den $(i - 1)$ -ten Eintrag hat, also der Knoten mit Schlüssel 1 vom Algorithmus als erstes berücksichtigt wird usw.

a)



b)



Lösung: _____

a) Der gegebene Graph hat die folgende topologische Sortierung:

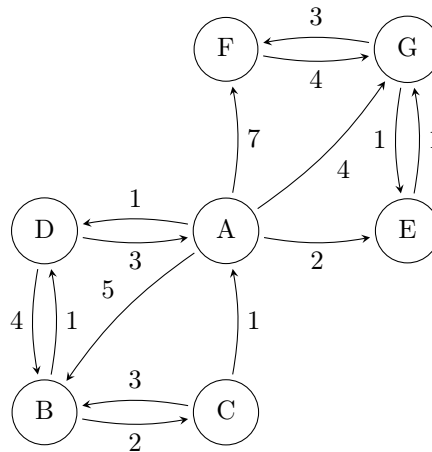
2(1), 9(2), 5(3), 4(4), 8(5), 7(6), 6(7), 3(8), 1(9)

b) Der gegebene Graph hat die folgende topologische Sortierung:

2(1), 8(2), 9(3), 20(4), 15(5), 14(6), 18(7), 13(8), 7(9), 3(10), 17(11), 21(12), 22(13), 16(14), 10(15), 6(16), 5(17), 4(18), 1(19), 12(20), 11(21), 19(22)

Tutoraufgabe 5 (Dijkstra):

Betrachten Sie den folgenden Graphen:



Führen Sie den Dijkstra Algorithmus auf diesem Graphen mit dem Startknoten A aus. Füllen Sie dazu die nachfolgende Tabelle aus, indem Sie den Wert von v und key nach jeder Iteration der `while`-Schleife eintragen:

v	A					
key[A]						
key[B]						
key[C]						
key[D]						
key[E]						
key[F]						
key[G]						

Lösung: _____

v	A	D	E	G	B	F
key[A]	0	0	0	0	0	0
key[B]	5	5	5	5	5	5
key[C]	∞	∞	∞	∞	7	7
key[D]	1	1	1	1	1	1
key[E]	2	2	2	2	2	2
key[F]	7	7	7	6	6	6
key[G]	4	4	3	3	3	3