Algorithmen und Datenstrukturen 2
- 1. Seminar -

Dominic Rose
Bioinformatics Group, University of Leipzig

Sommersemester 2010

Outline

1. Übungsserie: 3 Aufgaben, insgesamt ~~30~~ 28 Punkte

A1 Spannbäume (~~10~~ 8 Punkte)

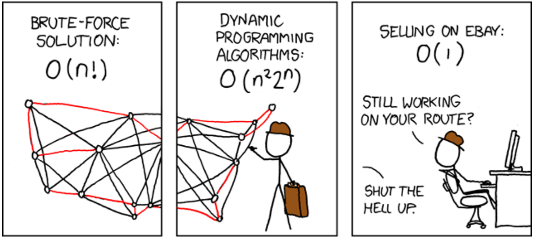
A2 Graphen, Eigenschaften (11 Punkte)

A3 Graphen, Traversierung (9 Punkte)

Vorlesung ADS2

Themen:

1. Graphenalgorithmen
2. Verarbeitung von Zeichenketten:
   Suche, Vergleich, Kompression
3. Dynamische Programmierung



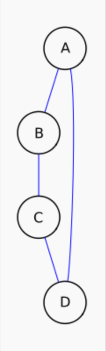
Wiederholung: Graphen

Ein Tupel $G = (V, E)$ heißt (ungerichteter) Graph, wenn V eine endliche Menge (von Knoten) und E eine Menge von (ungeordneten) Paaren von Elementen in V ist (Kanten).

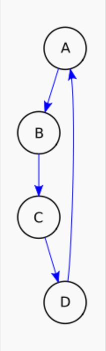
Wiederholung: Graphen

Ein Graph G ist ein Tupel (V,E), wobei V eine Menge von *Knoten* (englisch *vertex/vertices*, oft auch *Ecken* genannt) und E eine Menge von *Kanten* (engl. *edge/edges*, manchmal auch *Bögen* genannt) bezeichnet. Dabei ist E in

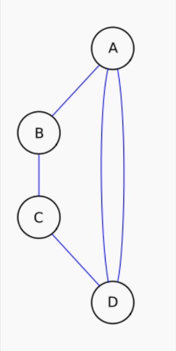
- *ungerichteten Graphen ohne Mehrfachkanten* eine Teilmenge aller 2-elementigen Teilmengen von V,
- *gerichteten Graphen ohne Mehrfachkanten* eine Teilmenge des kartesischen Produktes $V \times V$,
- *ungerichteten Graphen mit Mehrfachkanten* eine Multimenge über der Menge aller 2-elementigen Teilmengen von V,
- *gerichteten Graphen mit Mehrfachkanten* eine Multimenge über dem kartesischen Produkt $V \times V$,
- *Hypergraphen* eine Teilmenge der Potenzmenge von V.



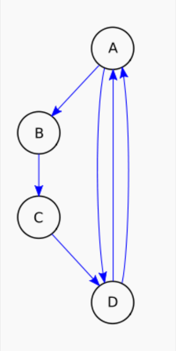
1. ungerichteter Graph ohne Mehrfachkanten



2. gerichteter Graph ohne Mehrfachkanten



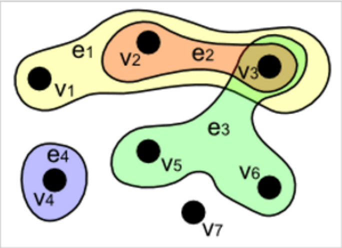
3. ungerichteter Graph mit Mehrfachkanten



4. gerichteter Graph mit Mehrfachkanten

Wiederholung: Graphen

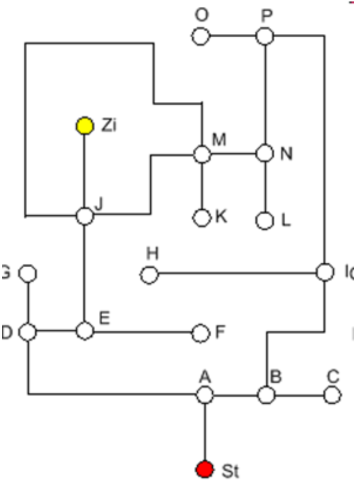
Hypergraph:



Sample of hypergraph:
$X = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7\}$,
$E = \{e_1, e_2, e_3, e_4\}$
$= \{\{v_1, v_2, v_3\}, \{v_2, v_3\},$
$\{v_3, v_5, v_6\}, \{v_4\}\}$.

Wiederholung: Graphen

Motivation:



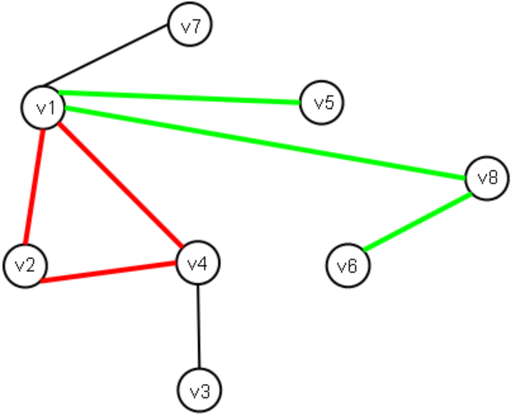
Wiederholung: Graphen

Terminologie:

- Weg, Pfad, Zyklus, Zusammenhang, Bäume, Wälder
- Teilgraph, spannender Teilgraph, induzierter Teilgraph

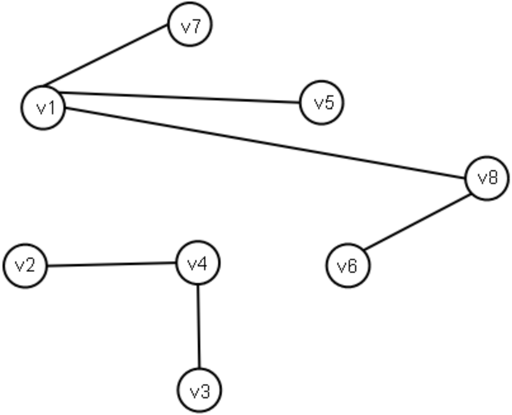
Wiederholung: Graphen

Weg, Pfad, Zyklus, Zusammenhang, Bäume, Wälder:



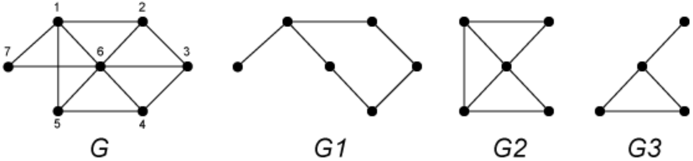
Wiederholung: Graphen

Weg, Pfad, Zyklus, Zusammenhang, Bäume, Wälder:



Wiederholung: Graphen

Teilgraph, spannender Teilgraph, induzierter Teilgraph:

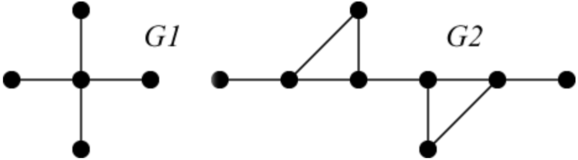


G1, G2, G3 sind Teilgraphen von G,
nur G2 und G3 sind induzierte Teilgraphen.

(spannender Teilgraph: Die beiden Knotenmengen stimmen überein)

Wiederholung: Graphen

Neu: Minor.



Ein Graph G_1 wird Minor des Graphen G_2 genannt, falls G_1 isomorph aus einem Untergraphen von G_2 durch Knotenverschmelzung entsteht.

Wiederholung: Graphen

Handshaking-Lemma:

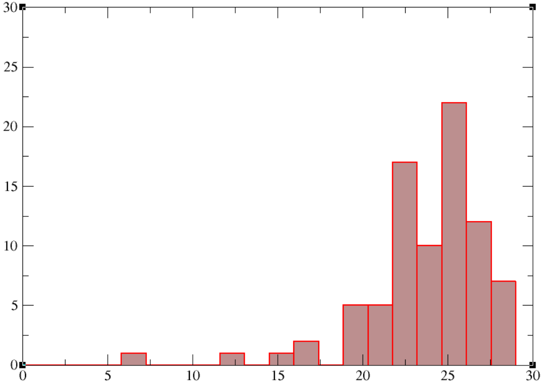
- Auf einer Party begrüßen sich alle Gäste per Händedruck, jeder merkt sich wie oft er bzw. sie jmd. die Hand gegeben hat. Wie oft wurden insgesamt Hände geschüttelt?
- Graphentheorie: Vollständiger Graph, jeder Knoten weist eine Kante zu jedem anderen auf.
- Frage: Wie groß ist die Summe über alle Knoten-Grade.
- Handshaking-Lemma: $\sum_{v \in V} d(v) = 2 \cdot |E|$

  (Summe über die Knoten-Grade ist stets gleich der doppelten Zahl von Kanten.)
- Beweis?

Wiederholung: Graphen

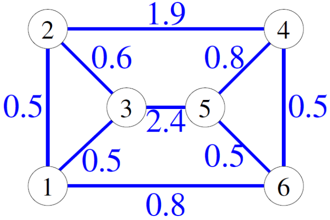
- Handshaking-Lemma: $\sum_{v \in V} d(v) = 2 \cdot |E|$
- Beweis: Induktion über V
- IA: sei $n = 1$, in Graph mit 1 Knoten existiert entweder 0 oder 1 Kante (Schleife), stimmt.
- IS: $n \longrightarrow n + 1$, Behauptung gelte für induzierten Teilgraph mit n Knoten. Sei k die Zahl der Kanten, die v_{n+1} berühren. Jede dieser Kanten erhöht jeweils im Start- und Zielknoten den Grad um 1. Somit wird die Summe über alle Grade um $2k$ erhöht.

Serie 1



Aufgabe 1

Gegeben sei der folgende ungerichtete gewichtete Graph G.



(a) Finden Sie einen minimalen Spannbaum von G mit dem Algorithmus von Kruskal. Schreiben Sie als Resultat die Kanten des Baums in der Reihenfolge hin, in der sie hinzugefügt werden. (4 Punkte)

Spannbaum?

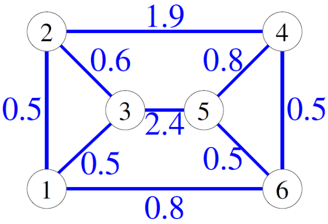
- Geg.: Gewichteter zusammenhängender Graph $G = (V, E, w)$ mit $|V| \geq 2$.
- Ges.: Spannbaum $T = (V, F)$ mit min. Kantensumme. Wähle also die Kantenmenge $F \subseteq E$ so, daß

$$\sum_{e \in F} w(e)$$

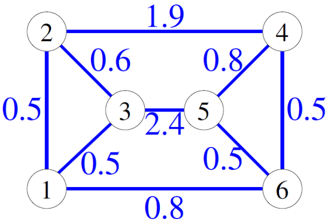
möglichst klein wird.

$\longrightarrow$ Kruskal-Algorithmus.

Aufgabe 1

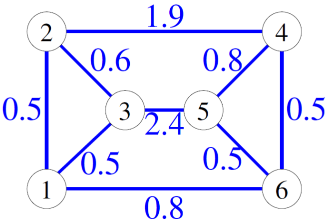


Aufgabe 1



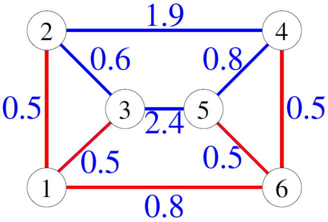
- Warteschlange $Q = [\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{2,3\},\{1,6\},\{4,5\},\{2,4\},\{3,5\}]$

Aufgabe 1



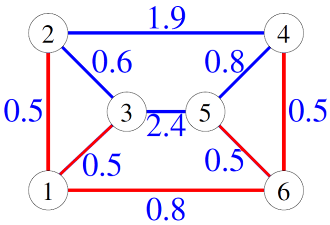
- Warteschlange $Q =$ $[\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{2,3\},\{1,6\},\{4,5\},\{2,4\},\{3,5\}]$
- $T = (V, F)$,
  Kanten $F = \{\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{1,6\}\}$

Aufgabe 1



$$F = \{\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{1,6\}\}$$

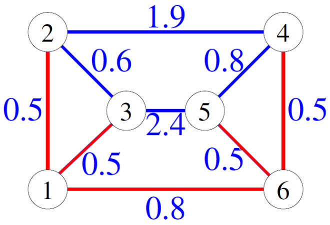
Aufgabe 1



$$F = \{\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{1,6\}\}$$

(b) Ist der gefundene min. Spannbaum eindeutig?. (1 Punkt)

Aufgabe 1



$$F = \{\{1,2\},\{1,3\},\{5,6\},\{4,6\},\{1,6\}\}$$

(b) Ist der gefundene min. Spannbaum eindeutig?. (1 Punkt)
JA :-)

Aufgabe 1

(c) Von einem ungerichteten gewichteten zusammenhängenden Graphen $G = (V, E, w)$ sei nur die Anzahl der Knoten $n = |V|$ und die Menge der benutzten Kantengewichte

$$B = \{w(e) | e \in E\}$$

bekannt. Geben Sie ein hinreichendes Kriterium dafür an, daß G einen eindeutigen minimalen Spannbaum hat. Das Kriterium soll außer B und n keine weitere Information über den Graphen verwenden. (2 Punkte)

Aufgabe 1

(c) Von einem ungerichteten gewichteten zusammenhängenden Graphen $G = (V, E, w)$ sei nur die Anzahl der Kanten $m = |E|$ und die Menge der benutzten Kantengewichte

$$B = \{w(e) | e \in E\}$$

bekannt. Geben Sie ein hinreichendes Kriterium dafür an, daß G einen eindeutigen minimalen Spannbaum hat. Das Kriterium soll außer B und m keine weitere Information über den Graphen verwenden. (2 Punkte)

Aufgabe 1

Fehler in der Aufgabenstellung: Gemeint war, dass die Anzahl der Kanten $m = |E|$ bekannt ist und (wie gegeben) die Gewichtsmenge B. Dann folgt aus $|B| = m$, dass der minimale Spannbaum eindeutig ist. Die reverse Implikation gilt nicht, der Graph aus 1a ist ein Gegenbeispiel.

Aufgabe 1

Fehler in der Aufgabenstellung: Gemeint war, dass die Anzahl der Kanten $m = |E|$ bekannt ist und (wie gegeben) die Gewichtsmenge B. Dann folgt aus $|B| = m$, dass der minimale Spannbaum eindeutig ist. Die reverse Implikation gilt nicht, der Graph aus 1a ist ein Gegenbeispiel.

Will man die Aufgabe lösen wie gegeben, so kann man z.B. für den Spezialfall eines vollständigen Graphen doch noch ein hinreichendes Kriterium angeben: Ist $|B| = n(n-1)/2$, dann sind die Kantengewichte auch paarweise verschieden und deshalb der minimale Spannbaum eindeutig.

Aufgabe 1

(d) Der Algorithmus von Kruskal werde auf einen nicht-zusammenhängenden gewichteten Graphen G mit n Knoten angewendet und liefere eine Kantenmenge T mit $r = |T|$ Kanten. Ist (V, T) ein Spannbaum von G? Welche Information über G entnehmen Sie r und n? (3 Punkte)

Aufgabe 1

(d) Der Algorithmus von Kruskal werde auf einen nicht-zusammenhängenden gewichteten Graphen G mit n Knoten angewendet und liefere eine Kantenmenge T mit $r = |T|$ Kanten. Ist (V, T) ein Spannbaum von G? Welche Information über G entnehmen Sie r und n? (3 Punkte)

$\longrightarrow$ (V, T) ist kein Spannbaum, da nicht zusammenhängend. $n - r$ ist die Zahl der Zusammehangskomponenten von G.

Aufgabe 2

Der folgende gerichtete Graph sei durch seine Kantenliste gegeben:

$$7, 10, 1, 7, 1, 4, 1, 5, 2, 1, 2, 3, 3, 6, 6, 5, 4, 5, 5, 2, 7, 4$$

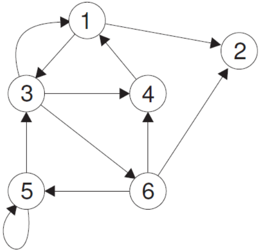
(a) Geben Sie die Adjazenzmatrix des Graphen an. (3 Punkte)

Wiederholung: Speicherung v. G.

- Knoten-/Kantenliste
- Adjazenzmatrix (Inzidenzmatrix)
- Adjazenzliste

Wiederholung: Knotenliste

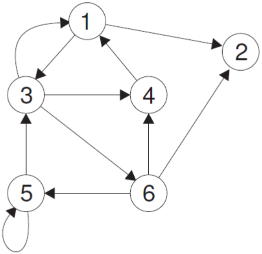
- Erstes Element: Anzahl der Knoten
- Zweites Element: Anzahl der Kanten
- Danach für jeden Knoten: Codierung der Anzahl ausgehender Kanten *n* und der *n* Zielknoten



6, 11, 2, 2, 3, 0, 3, 1, 4, 6, 1, 1, 2, 3, 5, 3, 2, 4, 5

Wiederholung: Kantenliste

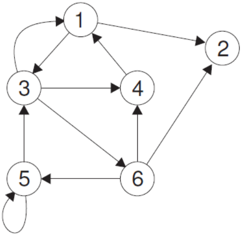
- Erstes Element: Anzahl der Knoten
- Zweites Element: Anzahl der Kanten
- Alle weiteren Wertepaare: Kanten-Codierungen



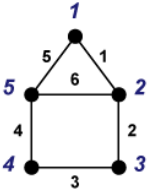
6, 11, 1, 2, 1, 3, 3, 1, 4, 1, 3, 4, 3, 6, 5, 3, 5, 5, 6, 5, 6, 2, 6, 4

Wiederholung: Adjazenzmatrix

$$G_g = \begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$$

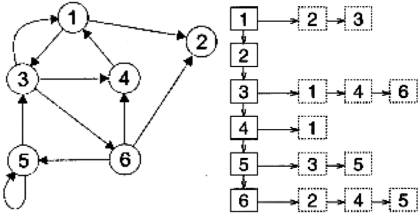


Wiederholung: Inzidenzmatrix



$$\Leftrightarrow \quad \begin{array}{cc} \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 & e_6 \end{matrix} & \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix} & \begin{matrix} v_1 \\ v_2 \\ v_3 \\ v_4 \\ v_5 \end{matrix} \end{array}$$

Wiederholung: Adjazenzliste



Aufgabe 2

Der folgende gerichtete Graph sei durch seine Kantenliste gegeben:

$$7, 10, 1, 7, 1, 4, 1, 5, 2, 1, 2, 3, 3, 6, 6, 5, 4, 5, 5, 2, 7, 4$$

(a) Geben Sie die Adjazenzmatrix des Graphen an. (3 Punkte)

Aufgabe 2

Der folgende gerichtete Graph sei durch seine Kantenliste gegeben:

$$7, 10, 1, 7, 1, 4, 1, 5, 2, 1, 2, 3, 3, 6, 6, 5, 4, 5, 5, 2, 7, 4$$

(a) Geben Sie die Adjazenzmatrix des Graphen an. (3 Punkte)

$$\begin{pmatrix} 0 & 0 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{pmatrix}$$

Aufgabe 2

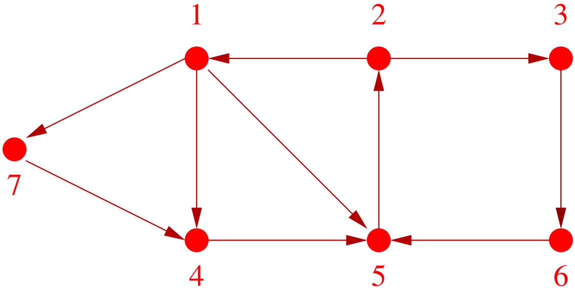
(b) Zeichnen Sie den Graphen. (2 Punkte)

$$7, 10, 1, 7, 1, 4, 1, 5, 2, 1, 2, 3, 3, 6, 6, 5, 4, 5, 5, 2, 7, 4$$

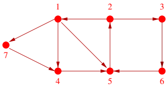
Aufgabe 2

(b) Zeichnen Sie den Graphen. (2 Punkte)

$$7, 10, 1, 7, 1, 4, 1, 5, 2, 1, 2, 3, 3, 6, 6, 5, 4, 5, 5, 2, 7, 4$$

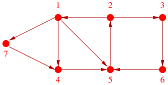


Aufgabe 2



(c) Besitzt dieser Graph einen Hamiltonschen Zyklus? Falls ja: Geben Sie einen an. Falls nein: Begründen Sie dies möglichst kurz. (2 Punkte)

Aufgabe 2

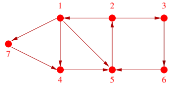


(c) Besitzt dieser Graph einen Hamiltonschen Zyklus? Falls ja: Geben Sie einen an. Falls nein: Begründen Sie dies möglichst kurz. (2 Punkte)

$\longrightarrow$ Nein, es gibt keinen Hamiltonschen Zyklus in diesem Graphen. Nehmen wir an, es gäbe einen Hamiltonschen Zyklus h. Der Vorgänger des Knotens 3 in h ist in jedem Falle der Knoten 2, denn anders wird 3 nicht erreicht. Direkt nach 3 besucht h den Knoten 6, dann 5 und dann 2. Der betrachtete Kreis ist also $(2, 3, 6, 5)$ im Widerspruch zu der Annahme, dass wir es mit einem Hamiltonschen Kreis zu tun haben.

(Hamiltonscher Zyklus: Geschlossener Weg, der genau 1x durch jeden Knoten führt.)

Aufgabe 2

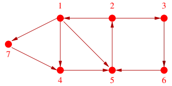


(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

Aufgabe 2



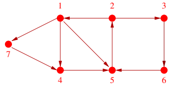
(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

	(2,1,5,2,1)	(6,5,4,1)	(2,1,5,6)	(2,1,4,5,2)
Kantenfolge	?	?	?	?
Kantenzug	?	?	?	?
Pfad	?	?	?	?
Zyklus	?	?	?	?

Aufgabe 2



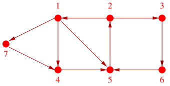
(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

	(2,1,5,2,1)	(6,5,4,1)	(2,1,5,6)	(2,1,4,5,2)
Kantenfolge	ja	nein	nein	ja

Aufgabe 2



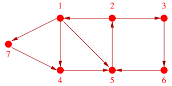
(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

	(2,1,5,2,1)	(6,5,4,1)	(2,1,5,6)	(2,1,4,5,2)
Kantenfolge	ja	nein	nein	ja
Kantenzug	nein	nein	nein	ja

Aufgabe 2



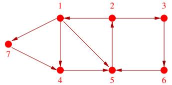
(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

	(2,1,5,2,1)	(6,5,4,1)	(2,1,5,6)	(2,1,4,5,2)
Kantenfolge	ja	nein	nein	ja
Kantenzug	nein	nein	nein	ja
Pfad	nein	nein	nein	nein

Aufgabe 2



(d) Betrachten Sie die Knotenfolgen

(2,1,5,2,1), (6,5,4,1), (2,1,5,6), (2,1,4,5,2)

Geben Sie zu jeder Knotenfolge an, ob sie für den gegebenen Graphen eine Kantenfolge, ein Kantenzug, ein Pfad, ein Zyklus ist. (4 Punkte)

	(2,1,5,2,1)	(6,5,4,1)	(2,1,5,6)	(2,1,4,5,2)
Kantenfolge	ja	nein	nein	ja
Kantenzug	nein	nein	nein	ja
Pfad	nein	nein	nein	nein
Zyklus	nein	nein	nein	ja

Aufgabe 3

Führen Sie auf dem Graphen aus Aufgabe 2 folgende Traversierungen durch.

(a) einen Breitendurchlauf beginnend bei Knoten 7
(b) einen Breitendurchlauf beginnend bei Knoten 5
(c) einen Tiefendurchlauf beginnend bei Knoten 5

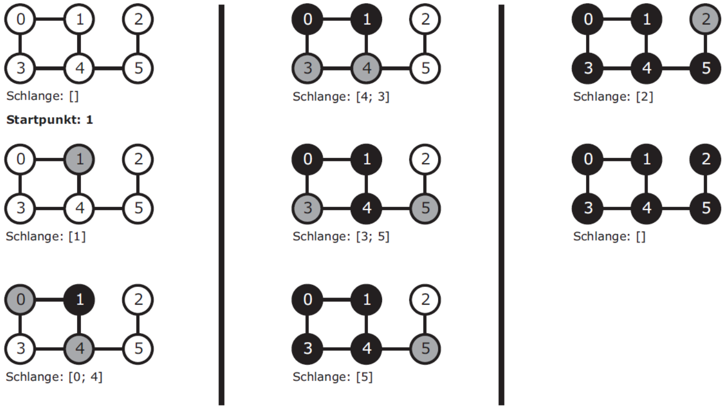
Geben Sie die Knoten des Graphen in der Reihenfolge an, in der sie bearbeitet werden. Bei Mehrdeutigkeit wird der Knoten mit dem kleinsten Index zuerst bearbeitet.

Wiederholung: Traversierung

<u>Breitensuche/-durchlauf</u> (Breadth First Search, BFS)

- ausgehend von Startknoten werden zunächst alle direkt erreichbaren Knoten bearbeitet
- danach die über mindestens zwei Kanten vom Startknoten erreichbaren Knoten, dann die über drei Kanten usw.
- es werden also erst die Nachbarn besucht, bevor zu den Söhnen gegangen wird.
- Bearbeite einen Knoten, der in n Schritten von u erreichbar ist, erst wenn alle Knoten abgearbeitet wurden, die in $n-1$ Schritten erreichbar sind.

Wiederholung: Breitensuche

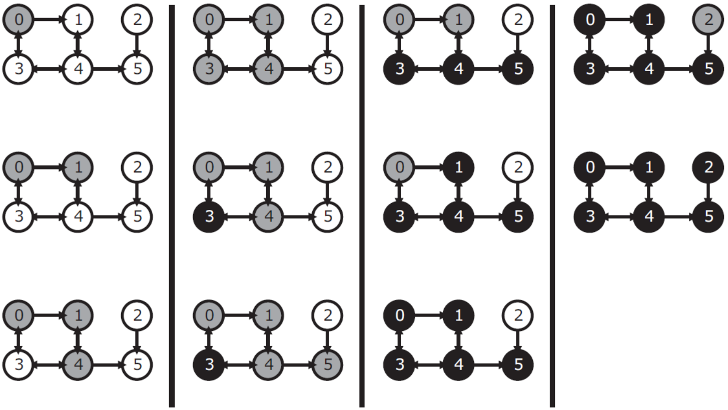


Wiederholung: Traversierung

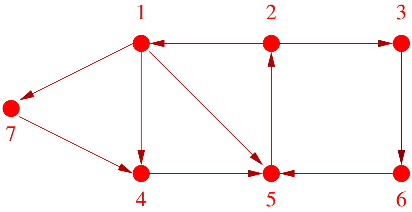
Tiefensuche/-durchlauf (Depth First Search, DFS)

- ausgehend von Startknoten werden zunächst rekursiv alle Söhne (Nachfolger) bearbeitet; erst dann wird zu den Nachbarn gegangen
- kann mit Stack-Datenstruktur für noch zu bearbeitende Knoten realisiert werden

Wiederholung: Tiefensuche

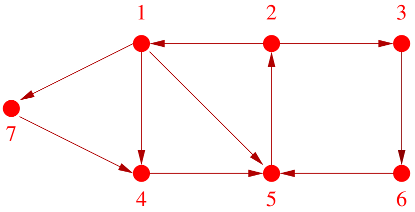


Aufgabe 3



(a) Breitendurchlauf, beginnend bei 7:

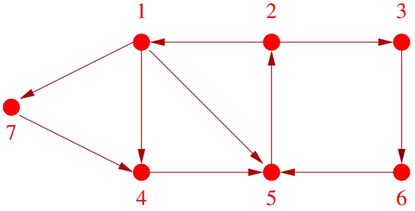
Aufgabe 3



(a) Breitendurchlauf, beginnend bei 7:
7 4 5 2 1 3 6

(b) Breitendurchlauf, beginnend bei 5:

Aufgabe 3

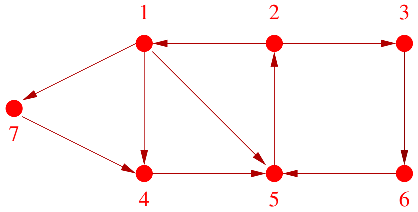


(a) Breitendurchlauf, beginnend bei 7:
7 4 5 2 1 3 6

(b) Breitendurchlauf, beginnend bei 5:
5 2 1 3 4 7 6

(c) Tiefendurchlauf, beginnend bei 5:

Aufgabe 3



(a) Breitendurchlauf, beginnend bei 7:
7 4 5 2 1 3 6

(b) Breitendurchlauf, beginnend bei 5:
5 2 1 3 4 7 6

(c) Tiefendurchlauf, beginnend bei 5:
5 2 1 4 7 3 6 (Ordnung nach in-Zeiten)
4 7 1 6 3 2 5 (Ordnung nach out-Zeiten)