

Algorithmen und Datenstrukturen
Klausur WS 2019/20
Angewandte Informatik Bachelor

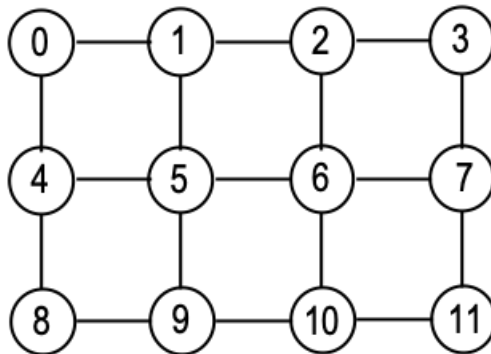
Name	
Matrikelnummer	

Aufgabe 1	Tiefen- und Breitensuche in ungerichteten Graphen	14	
Aufgabe 2	AVL-Bäume	9	
Aufgabe 3	B-Bäume	13	
Aufgabe 4	Algorithmus von Dijkstra	11	
Aufgabe 5	Tiefensuchbaum und Artikulationspunkte	13	
Summe		60	

Aufgabe 1 Tiefen- und Breitensuche in Graphen

(14 Punkte)

Ein $n*m$ -Manhattan-Graph ist ein ungewichteter Graph, deren Knoten und Kanten gitterförmig in der Ebene mit n Zeilen und m Spalten angeordnet werden können. Ein $3*4$ -Manhattan-Graph sieht beispielsweise wie folgt aus:

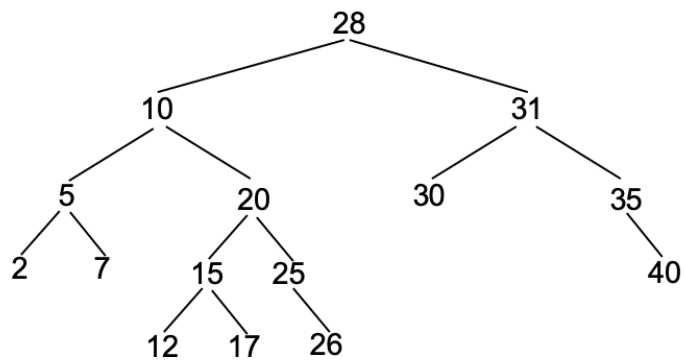


- a) Wieviel Knoten hat ein $n*m$ -Manhattan-Graph ganz allgemein (1 Punkt):
- b) Wieviel Kanten hat ein $n*m$ -Manhattan-Graph ganz allgemein (3 Punkte):
- c) Geben Sie für den oben abgebildeten Graphen die Reihenfolge der besuchten Knoten an, wenn der Graph mit Tiefensuche mit Startknoten 0 traversiert wird. Betrachten Sie die Nachbarn eines Knotens in der durch die Knotennummerierung gegebenen Reihenfolge (3 Punkte).
- d) Geben Sie für den oben abgebildeten Graphen die Reihenfolge der besuchten Knoten an, wenn der Graph mit Breitensuche mit Startknoten 0 traversiert wird. Betrachten Sie die Nachbarn eines Knotens in der durch die Knotennummerierung gegebenen Reihenfolge (3 Punkte).
- e) Welches Problem entsteht bei der rekursiven Tiefensuche bei einem sehr großen $n*m$ -Manhattan-Graph? Zwei kurze Sätze genügen (4 Punkte).

Aufgabe 2 AVL-Bäume

(9 Punkte)

Gegebener ist folgender AVL-Baum B:



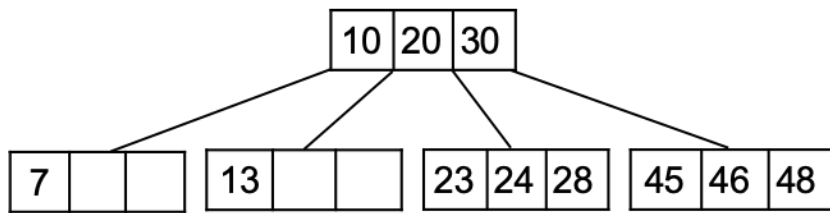
a) Fügen Sie im AVL-Baum B (siehe Abb. oben) die Zahl 18 ein. (4 Punkte)

b) Löschen Sie im AVL-Baum B (siehe Abb. oben) die Zahl 28. (5 Punkte)

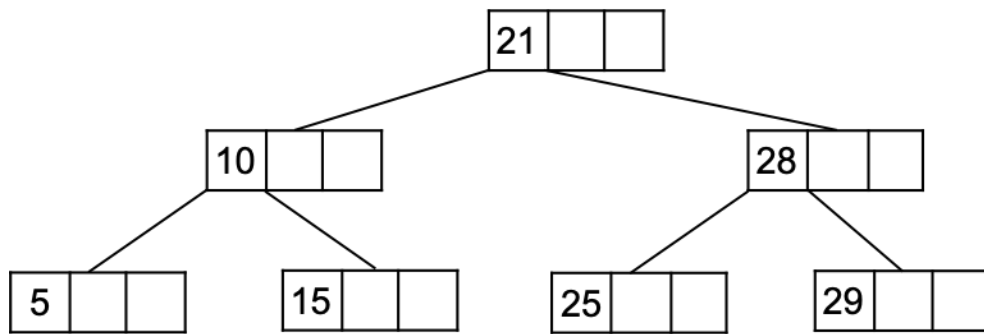
Aufgabe 3 B-Bäume

(13 Punkte)

a) Fügen Sie in folgendem B-Baum (der Ordnung 4) den Schlüssel 47 und dann 21 ein. (6 Punkte)



b) Löschen Sie in folgendem B-Baum (der Ordnung 4) den Schlüssel 21 und dann 15. (7 Punkte)



Aufgabe 4 Algorithmus von Dijkstra

(11 Punkte)

Ein gewichteter, gerichteter Graph mit der Knotenmenge $V = \{1, 2, 3, 4, 5, 6\}$ ist durch folgende Adjazenzmatrix gegeben. Bestimmen Sie mit dem Algorithmus von Dijkstra vom Startknoten $s = 5$ zu allen anderen Knoten jeweils einen günstigsten Weg.

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

- a) Tragen Sie in folgende Tabelle nach jedem Besuchsschritt folgendes ein:
- der besuchte Knoten b
 - die Kosten $d[v]$ für den günstigsten Weg von Startknoten s nach v
 - den Vorgängerknoten $p[v]$ für den günstigsten Weg von Startknoten s nach v .

Wichtig: Haben mehrere Kandidaten denselben d -Wert, dann wird der Kandidat mit kleinster Nummer als nächster Knoten besucht.

Hinweis: Es brauchen nur die d - und p -Werte eingetragen werden, die sich geändert haben. Die endgültigen p - und d -Werte können durch Umrandung besonders gekennzeichnet werden. (8 Punkte)

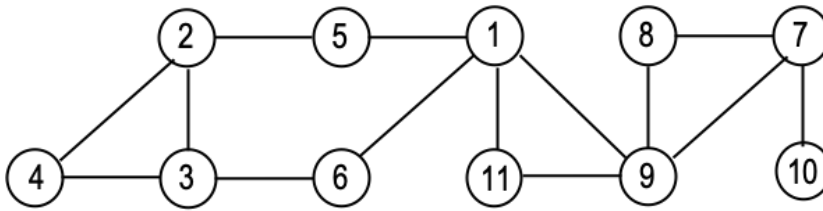
b	d[1]	d[2]	d[3]	d[4]	d[5]	d[6]	p[1]	p[2]	p[3]	p[4]	p[5]	p[6]

- b) Geben Sie den gefundenen günstigsten Weg von 5 nach 1 an. (2 Punkte)
- c) Welche Kosten hat der günstigste Weg von 5 nach 1? (1 Punkt)

Aufgabe 5 Tiefensuchbaum und Artikulationspunkte

(13 Punkte)

Gegeben sei folgender ungerichteter Graph:



- a) Geben Sie alle Artikulationspunkte an (2 Punkte).
- b) Geben Sie den Tiefensuchbaum mit Rückwärtskanten (TR) mit Wurzel 1 an. Betrachten Sie die Nachbarn eines Knotens in der durch die Knotennummerierung gegebenen Reihenfolge. Kennzeichnen Sie die Rückwärtskanten durch Beschriftung „R“ oder eine andere Farbe. (6 Punkte)
- c) Begründen Sie mit Hilfe des TR, warum Knoten 1 und 9 Artikulationspunkte (APe) und Knoten 3 kein Artikulationspunkt ist? Folgender Begriff darf verwendet werden: Ein Rückwärtsweg ist ein Weg in einem Tiefensuchbaum mit einer beliebig langen Folge von Vorwärtskanten und dann genau einer Rückwärtskante. (5 Punkte)