Technische Universität Braunschweig

Bitte so markieren: ☐ ☒ ☐ ☐ ☐ Bitte verwenden Sie einen Kugelschreiber oder nicht zu starken Filzstift. Dieser Fragebogen wird maschinell erfasst.

Korrektur: ☐ ■ ☐ ☒ ☐ Bitte beachten Sie im Interesse einer optimalen Datenerfassung die links gegebenen Hinweise beim Ausfüllen.

Bitte ausfüllen (Die Angabe des Namens ist freiwillig.):

Vorname: ______________________

Nachname: ______________________

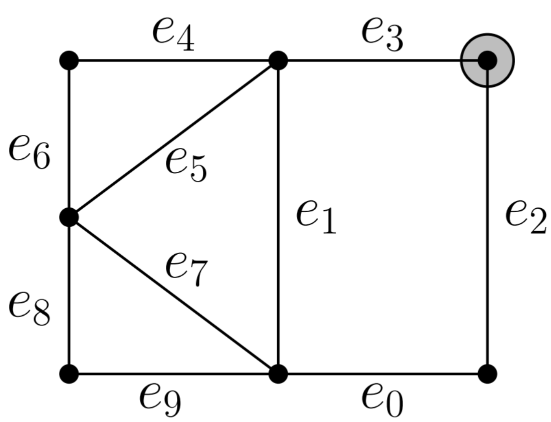
Für die eindeutige Zuordnung der Prüfung übertragen Sie bitte Ihre Prüfungsteilnehmer-ID gewissenhaft in die dafür vorgesehenen Felder. Alle Seiten sind vollständig individualisiert und nicht mit anderen Prüfungen tauschbar.

Prüfungsteilnehmer-ID für den Prüfungsbogen Nr.: 0:

0	☐	☐	☐	☐	☐	☐	☐
1	☐	☐	☐	☐	☐	☐	☐
2	☐	☐	☐	☐	☐	☐	☐
3	☐	☐	☐	☐	☐	☐	☐
4	☐	☐	☐	☐	☐	☐	☐
5	☐	☐	☐	☐	☐	☐	☐
6	☐	☐	☐	☐	☐	☐	☐
7	☐	☐	☐	☐	☐	☐	☐
8	☐	☐	☐	☐	☐	☐	☐
9	☐	☐	☐	☐	☐	☐	☐

1. Eulertouren (15 Punkte)

Betrachte folgenden Graphen.



Führe den Algorithmus von Fleury zum Finden einer Eulertour auf diesen Graphen aus. Starte dabei mit dem grau hinterlegten Knoten. Kommen zum einen Zeitpunkt mehrere Kanten in Frage, benutze diejenige Kante mit dem kleinsten Index.

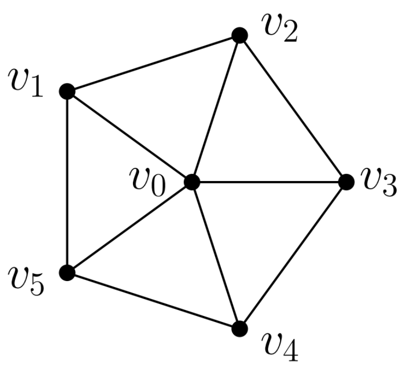
1.1 Gib die Indizes der Kanten in der abgelaufenen Reihenfolge an.

[ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ], [ ]

1.2 Gib zwei notwendige Bedingungen an, die ein Graph G erfüllen muss, damit eine Eulertour in G existieren kann.

1. Eulertouren (15 Punkte) [Fortsetzung]

Betrachte nun den nachfolgenden Graphen W_5.



1.3 Lösche drei Kanten aus W_5, sodass der resultierende Graph eine Eulertour enthält.
Gib die drei Kanten an.

1.4 Begründe kurz, warum es nicht reicht, weniger als drei Kanten zu löschen.

1.5 Zeige: Es können in den W_5 Kanten nicht so eingefügt werden, dass der resultierende (einfache) Graph eine Eulertour enthält.

2. Datenstrukturen - Stapel und Warteschlange (10 Punkte)

Betrachte die folgende, auf einem Array implementierte Warteschlange.

1	2	3	4	5	6	7	8
6	10	4	2	7	9	1	3
		ende[Q]			kopf[Q]		

2.1 Welche der folgenden Elemente (Felder 1 bis 4) sind in der Warteschlange enthalten? (Mindestens eine Antwort ist korrekt.)

- ☐ 6
- ☐ 10
- ☐ 4
- ☐ 2

2.2 Welche der folgenden Elemente (Felder 5 bis 8) sind in der Warteschlange enthalten? (Mindestens eine Antwort ist korrekt.)

- ☐ 7
- ☐ 9
- ☐ 1
- ☐ 3

Führe nun die Operation Enqueue(Q, 5) aus.

2.3 In welches Feld des Arrays wird die 5 eingetragen?

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8

2.4 Auf welches Feld zeigt der Zeiger ende[Q] nach der Enqueue-Operation?

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8

2.5 Auf welches Feld zeigt der Zeiger kopf[Q] nach der Enqueue-Operation?

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8

Führe die Operation Dequeue(Q) aus. Nimm dabei an, dass die vorherige Operation noch nicht ausgeführt wurde.

2.6 Welches Element wir durch die Dequeue-Operation zurückgegeben?

- ☐ 6
- ☐ 10
- ☐ 4
- ☐ 2
- ☐ 7
- ☐ 9
- ☐ 1
- ☐ 3

2.7 Auf welches Feld zeigt der Zeiger ende[Q] nach der Dequeue-Operation?

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8

2.8 Auf welches Feld zeigt der Zeiger kopf[Q] nach der Dequeue-Operation?

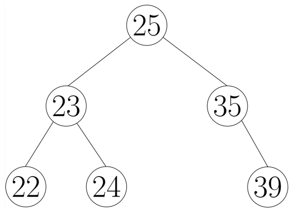
- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8

2. Datenstrukturen - Stapel und Warteschlange (10 Punkte) [Fortsetzung]

2.9 Warum müssen Elemente, die mittels Dequeue(Q) aus der Warteschlange entfernt werden, nicht explizit gelöscht werden?

3. Datenstrukturen - AVL-Bäume (12 Punkte)

Betrachte den folgenden AVL-Baum T_1. Führe die Operation Insert(T_1, 36) aus.



3.1 An welcher Stelle wird das neue Element (36) eingefügt?

- ☐ Links unter der 22
- ☐ Rechts unter der 22
- ☐ Links unter der 24
- ☐ Rechts unter der 24
- ☐ Links unter der 35
- ☐ Links unter der 39
- ☐ Rechts unter der 39

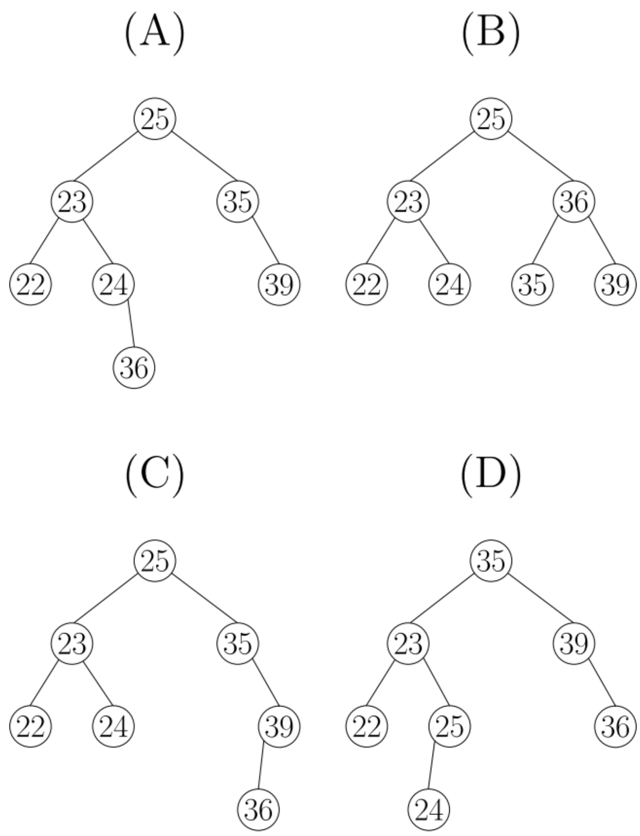
3.2 Welcher ist der niedrigste unbalancierte Knoten nach der Einfügeoperation, aber vor der Restructure-Operation?

- ☐ 22
- ☐ 23
- ☐ 24
- ☐ 25
- ☐ 35
- ☐ 39
- ☐ 36
- ☐ Suchbaum ist balanciert.

3.3 Wie viele Restructure-Operationen sind nötig, um den Suchbaum wieder zu balancieren?

- ☐ 0
- ☐ 1
- ☐ 2

Führe nun solange Restructure-Operationen aus, sodass wieder ein AVL-Baum entsteht. Betrachte folgende Suchbäume.

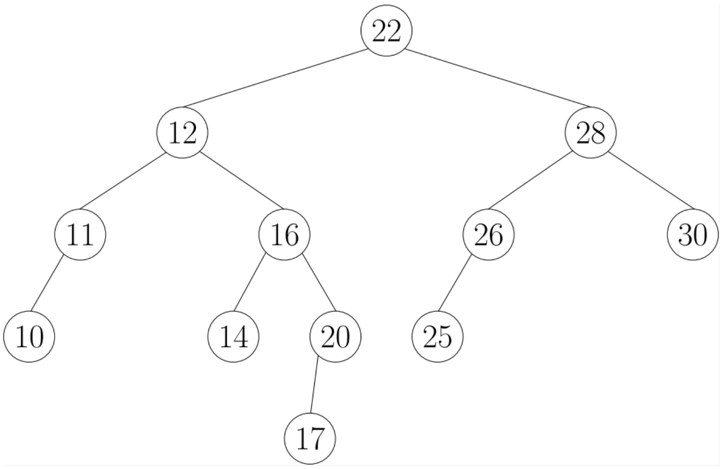


3.4 Welcher dieser Suchbäume ist das Resultat nach den Restructure-Operationen?

- ☐ (A)
- ☐ (B)
- ☐ (C)
- ☐ (D)

3. Datenstrukturen - AVL-Bäume (12 Punkte) [Fortsetzung]

Betrachte den folgenden AVL-Baum T_2. Führe die Operation Delete(T_2, 30) aus.



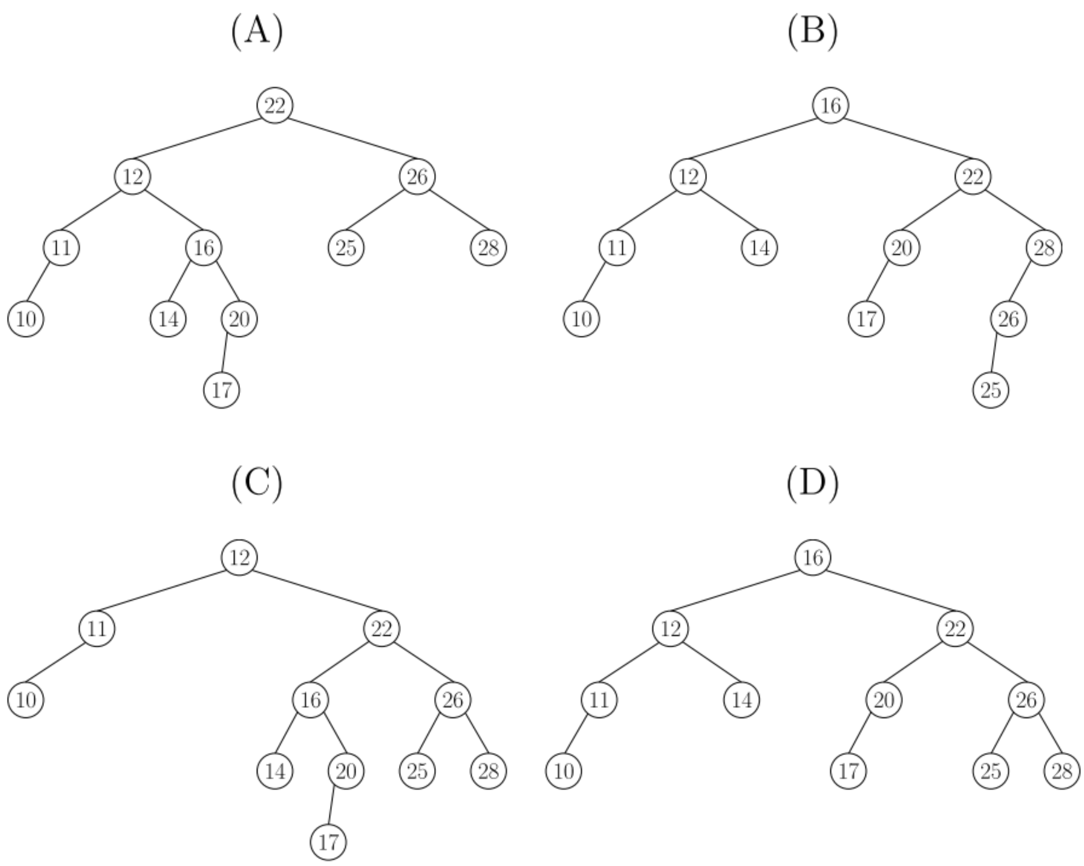
3.5 Welcher ist der niedrigste unbalancierte Knoten direkt nach der Löschoperation?

- ☐ 12
- ☐ 16
- ☐ 20
- ☐ 22
- ☐ 25
- ☐ 26
- ☐ 28
- ☐ Suchbaum ist balanciert.

3.6 Wie viele Restructure-Operationen sind notwendig, um den Suchbaum wieder zu balancieren?

- ☐ 0
- ☐ 1
- ☐ 2

Führe nun solange Restructure-Operationen aus, sodass wieder ein AVL-Baum entsteht. Betrachte folgende Suchbäume.



3.7 Welcher dieser Suchbäume ist das Resultat nach den Restructure-Operationen?

- ☐ (A)
- ☐ (B)
- ☐ (C)
- ☐ (D)

3. Datenstrukturen - AVL-Bäume (12 Punkte) [Fortsetzung]

3.8 Sei B ein voller binärer Suchbaum mit K inneren Knoten (also Knoten mit zwei Kindern). Zeige oder widerlege: B enthält genau K+1 Blätter (also Knoten von Grad 1).

4. Asymptotisches Wachstum (19 Punkte)

Betrachte die folgende Funktion.

$$f_1(n) := 3n(5\log(n) + 3)$$

4.1 In welcher der folgenden Klassen liegt f_1? (Mindestens eine Antwort ist korrekt.)

- ☐ O(n)
- ☐ O(n log n)
- ☐ O(n^2)
- ☐ In keiner der angegebenen Klassen

4.2 In welcher der folgenden Klassen liegt f_1? (Mindestens eine Antwort ist korrekt.)

- ☐ Ω(n)
- ☐ Ω(n log n)
- ☐ Ω(n^2)
- ☐ In keiner der angegebenen Klassen

Betrachte die folgende Funktion.

$$f_2(n) := 7n\sqrt{n}$$

4.3 In welcher der folgenden Klassen liegt f_2? (Mindestens eine Antwort ist korrekt.)

- ☐ O(n)
- ☐ O(n log n)
- ☐ O(n^2)
- ☐ In keiner der angegebenen Klassen

4.4 In welcher der folgenden Klassen liegt f_2? (Mindestens eine Antwort ist korrekt.)

- ☐ Ω(n)
- ☐ Ω(n log n)
- ☐ Ω(n^2)
- ☐ In keiner der angegebenen Klassen

Betrachte die folgende Funktion.

$$f_3(n) := \left(\frac{3}{2}\right)^n$$

4.5 In welcher der folgenden Klassen liegt f_3? (Mindestens eine Antwort ist korrekt.)

- ☐ O(n)
- ☐ O(n log n)
- ☐ O(n^2)
- ☐ In keiner der angegebenen Klassen

4.6 In welcher der folgenden Klassen liegt f_3? (Mindestens eine Antwort ist korrekt.)

- ☐ Ω(n)
- ☐ Ω(n log n)
- ☐ Ω(n^2)
- ☐ In keiner der angegebenen Klassen

---

4. Asymptotisches Wachstum (19 Punkte) [Fortsetzung]

Betrachte die folgende Funktion.

$$f_4(n) := n\sqrt{\log n}$$

4.7 In welcher der folgenden Klassen liegt f_4? (Mindestens eine Antwort ist korrekt.)
- ☐ O(n)
- ☐ O(n log n)
- ☐ $O(n^2)$
- ☐ In keiner der angegebenen Klassen

4.8 In welcher der folgenden Klassen liegt f_4? (Mindestens eine Antwort ist korrekt.)
- ☐ Ω(n)
- ☐ Ω(n log n)
- ☐ $\Omega(n^2)$
- ☐ In keiner der angegebenen Klassen

Im Folgenden betrachten wir nun Relationen zwischen den Klassen. Gib dazu in jeder Frage an, ob die Klasse A in B enthalten ist (aber nicht gleich), ob B in A enthalten ist (aber nicht gleich), ob die Klassen gleich sind (A = B), oder ob keines davon zutrifft.

4.9 Sei A = $O(3^n)$ und B = $O(2^n)$. Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

4.10 Sei A = $O(n^2)$ und B = Ω(n). Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

4.11 Sei A = $O(n^2)$ und B = Θ(1). Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

4.12 Sei A = Ω(n log n) und B = $\Omega(n^2)$. Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

4.13 Sei A = O(n log n) und B = Θ(log(n!)). Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

4.14 Sei A = $O(4n^2)$ und B = O(n*(3n-2)). Wie stehen A und B zueinander?
- ☐ A = B
- ☐ A Teilmenge von B
- ☐ B Teilmenge von A
- ☐ Keine der Antworten

---

4. Asymptotisches Wachstum (19 Punkte) [Fortsetzung]

Betrachte einen Algorithmus A, welcher alle Paare von n Elementen bearbeitet. Pro Paar benötigt A eine Laufzeit von $n^2 + 2n$.

Da die Bearbeitung unabhängig geschehen kann, darf parallelisiert werden. Dazu stehen acht Prozessoren bereit. Es können also acht Paare gleichzeitigt gearbeitet werden.

Alternativ können die Paare vorverarbeitet werden (Preprocessing), was insgesamt eine Laufzeit von $n^3 + n^2 \log(n)$ benötigt. Das Bearbeiten jedes Paares wird dadurch auf $n \log (n) + n$ reduziert.

4.15 Welche der drei Optionen (Nur A, A mit Parallelisierung oder mit Preprocessing) ist für große Eingabemengen am effizientesten? Begründe deine Wahl.

5. Master-Theorem (12 Punkte)

A - Betrachte die folgende Rekursionsgleichung.

$$T(n) := 2 \cdot T\left(\frac{n}{2}\right) + 3 \cdot T\left(\frac{n}{3}\right) + n^2 - 2n$$

5.1 Gib die im Master-Theorem auftretenden Parameter an.

m = ___, k = ___, α1 = α2 = ___/___, α3 = α4 = α5 = ___/___

5.2 Die Summe der α_i hoch k ist...

☐ < 1 ☐ = 1 ☐ > 1

5.3 Welches asymptotische Wachstum ergibt sich mit dem Master-Theorem für T(n)?

☐ $\Theta(n^2)$ ☐ $\Theta(n^2 \log n)$ ☐ $\Theta(n^3)$

B - Betrachte die folgende Rekursionsgleichung.

$$U(n) := U\left(\frac{4n}{5}\right) + U\left(\frac{3n}{5}\right) + n \log n + n^2$$

5.4 Gib die im Master-Theorem auftretenden Parameter an.

m = ___, k = ___, α1 = ___/___, α2 = ___/___

5.5 Die Summe der α_i hoch k ist...

☐ < 1 ☐ = 1 ☐ > 1

5.6 Welches asymptotische Wachstum ergibt sich mit dem Master-Theorem für T(n)?

☐ $\Theta(n^2)$ ☐ $\Theta(n^2 \log n)$ ☐ $\Theta(n^3)$

C - Betrachte die folgende Rekursionsgleichung.

$$V(n) := 81 \cdot V\left(\frac{n}{9}\right) + 3n - 12$$

5.7 Gib die im Master-Theorem auftretenden Parameter an.

m = ______, k = ___, α1 = ... = α81 = ___/___

5.8 Die Summe der α_i hoch k ist...

☐ < 1 ☐ = 1 ☐ > 1

5.9 Welches asymptotische Wachstum ergibt sich mit dem Master-Theorem für T(n)?

☐ $\Theta(n)$ ☐ $\Theta(n^2)$ ☐ $\Theta(n^3)$

---

5. Master-Theorem (12 Punkte) [Fortsetzung]

D - Betrachte die folgende Rekursionsgleichung.

$$S(n) := \sum_{i=1}^{4} S\left(\frac{n}{i}\right) + \sqrt{n}$$

5.10 Kann auf S(n) das Mastertheorem angewendet werden? Begründe deine Antwort.

6. Sortieren (9 Punkte)

Betrachte das folgende Array A.

$$A := [1 \quad 5 \quad 7 \quad 9 \quad 8 \quad 2 \quad 4 \quad 3]$$

Sortiere das Array A mit Hilfe von Quicksort. Wähle dazu in Partition das Pivotelement wie in der Vorlesung. Notiere dabei die Ergebnisse der Partition-Aufrufe, sofern sich das Array in dem jeweiligen Partition-Aufruf ändert. Notiere außerdem, wie viele Tauschoperationen (nicht mit der eigenen Position) durchgeführt wurden.

6.1 Gib das Array und das genutzte Pivotelement nach dem ersten ändernden Partition-Aufruf an.

, Pivotelement:

6.2 Wie viele ändernde Partition-Aufrufe wurden durchgeführt?

- ☐ 1
- ☐ 2
- ☐ 3
- ☐ 4
- ☐ 5
- ☐ 6

6.3 Wie viele Tauschoperationen wurden durchgeführt? (Dabei werden Tausche mit der eigenen Position nicht mitgezählt.)

- ☐ maximal 4
- ☐ 5
- ☐ 6
- ☐ 7
- ☐ 8
- ☐ mindestens 9

6.4 Es ist bekannt, dass Quicksort aus der Vorlesung im Worst-Case eine Laufzeit von $\Theta(n^2)$ besitzt.
Zeige oder widerlege: Es kann in Partition immer so ein Pivotelement gewählt werden, dass für Quicksort immer eine Laufzeit von $\Theta(n \log n)$ garantiert werden kann.

7. Mediane (11 Punkte)

7.1 Welche der folgenden Zahlen sind ein Median der Menge X = {2, 3, 7, 8, 10, 70, 100, 600, 1000}?
(Mindestens eine Antwort ist korrekt.)

- ☐ 2
- ☐ 3
- ☐ 7
- ☐ 8
- ☐ 10
- ☐ 70
- ☐ 100
- ☐ 600
- ☐ 1000

Betrachte den aus der Vorlesung bekannten Algorithmus zum Finden eines Rang-k-Elements.
Analysiere die Laufzeit des Algorithmus, wenn Dreiergruppen statt Fünfergruppen verwendet werden.

7.2 Die Laufzeit lässt sich rekursiv Darstellen als $T(n) = T(a*n) + T(b*n) + \Theta(n)$.
Gib die Werte für a und b als Quotienten an.

a = ___ / ___ , b = ___ / ___

7.3 Welche Laufzeit ergibt sich damit für den Algorithmus?

- ☐ $\Theta(1)$
- ☐ $\Theta(n)$
- ☐ $\Theta(n \log n)$
- ☐ $\Theta(n^2)$
- ☐ $\Theta(n^3)$

Analysiere nun die Laufzeit des Algorithmus, wenn Siebenergruppen statt Fünfergruppen verwendet werden.

7.4 Die Laufzeit lässt sich rekursiv Darstellen als $T(n) = T(a*n) + T(b*n) + \Theta(n)$.
Gib die Werte für a und b als Quotienten an.

a = ___ / ___ , b = ___ / ___

7.5 Welche Laufzeit ergibt sich damit für den Algorithmus?

- ☐ $\Theta(1)$
- ☐ $\Theta(n)$
- ☐ $\Theta(n \log n)$
- ☐ $\Theta(n^2)$
- ☐ $\Theta(n^3)$

Betrachte nun folgendes Problem.
Gegeben sind eine Menge X mit n Zahlen und ein m aus {1,...,n}.
Gesucht sind die m **kleinsten** Zahlen in X.

7.6 Beschreibe wie dieses Problem in O(n) Zeit gelöst werden kann.
(Hinweis: Es ist kein Pseudocode erforderlich.)

8. Kurzfragen (12 Punkte)

Betrachte folgende Kurzfragen. In jeder Teilaufgabe ist mindestens eine Antwort korrekt.

8.1 Breitensuche...
- ☐ ...benötigt O(n log n) Zeit.
- ☐ ...benutzt einen Stapel als Datenstruktur.
- ☐ ...findet kürzeste Wege in ungewichteten Graphen.

8.2 Jeder Hamiltonpfad ist...
- ☐ ...eine Kantenfolge.
- ☐ ...ein Weg.
- ☐ ...ein Pfad.

8.3 In einem Baum mit mindestens zwei Knoten...
- ☐ ...gibt es mindestens zwei Blätter (d.h. Knoten von Grad 1)
- ☐ ...ist der höchste Knotengrad drei.
- ☐ ...ist jeder Pfad zwischen zwei Knoten eindeutig.

8.4 In doppelt-verketteten Listen...
- ☐ ...dauert die Einfügeoperation O(1) Zeit.
- ☐ ...dauert das Suchen nach einem Element O(log n) Zeit.
- ☐ ...besitzt jedes Element je einen Zeiger auf Vorgänger und Nachfolger.

8.5 Welcher der folgenden Sortieralgorithmen besitzen eine Laufzeit von O(n log n)?
- ☐ Quicksort
- ☐ Mergesort
- ☐ Radixsort

8.6 Welche der folgenden Graphenprobleme lassen sich in linearer Zeit, d.h. O(n+m), lösen?
- ☐ Test auf Zusammenhang
- ☐ Test auf Kreisfreiheit
- ☐ Finden einer Eulertour