

Beantworten Sie die Fragen in Aufgabe 1 durch Ankreuzen; pro Frage genau ein Kreuz. Ist Ihre Antwort richtig, erhalten Sie einen Punkt. Ist sie falsch, wird Ihnen ein Punkt abgezogen. Machen Sie kein Kreuz, bleibt die Punktzahl unverändert. Sie erhalten mindestens 0 Punkte.

Aufgabe 1 (10 Punkte)

	Frage	richtig	falsch
1.	Redundanz bedeutet, dass dieselben Informationen in unterschiedlichen Versionen gespeichert werden	<input type="radio"/>	<input type="radio"/>
2.	Durch Konsolidierung erhält man ein redundanzfreies globales Schema	<input type="radio"/>	<input type="radio"/>
3.	Das Gridfile ist eine zweidimensionale Suchstruktur mit variabler Gittergröße	<input type="radio"/>	<input type="radio"/>
4.	Beim Schreiben von Log-Informationen gilt das WAL-Prinzip	<input type="radio"/>	<input type="radio"/>
5.	Ein Schedule heißt serialisierbar, wenn alle Schritte einer Transaktion hintereinander ablaufen	<input type="radio"/>	<input type="radio"/>
6.	Ein <code>unique foreign key</code> modelliert eine 1:1-Beziehung	<input type="radio"/>	<input type="radio"/>
7.	Ein Relationenschema ist in 2. Normalform, wenn jedes Nichtprimär-Attribut funktional abhängig ist von jedem Schlüsselkandidaten	<input type="radio"/>	<input type="radio"/>
8.	Mit der <code>check</code> -Klausel wird in SQL statische Integrität hergestellt	<input type="radio"/>	<input type="radio"/>
9.	Beim offenen Hashing werden die Datensätze in sortierter Reihenfolge gespeichert	<input type="radio"/>	<input type="radio"/>
10.	Das Zeitstempelverfahren zur Vermeidung von Deadlocks verwendet die Time-out-Strategie	<input type="radio"/>	<input type="radio"/>

Aufgabe 2 (10 Punkte)

Gegeben seien folgende Entity- und Relationship-Typen:

Entity-Typ	Attribute
Städte	PLZ, Name
Stadien	Name, Sitzplätze, Stehplätze
Vereine	Name, Anzahl_Titel, Vereinsmitglieder
Ligen	Name, Anzahl_Mannschaften
Spieler	Name, Tore, PassNr

Relationship-Typ
liegen_in
spielen_in
nehmen_teil
spielen_für
sind_untergeordnet

und folgende Beziehungen:

1. In einer Liga spielen immer 12 - 20 Mannschaften.
2. Ein Spieler darf in einer Saison höchstens für einen Verein spielen.
3. Ein Verein darf nur in maximal einem Stadion spielen.
4. In einem Stadion können mehrere Vereine spielen.
5. In einer Stadt können viele Stadien liegen.
6. Da ein Verein aus mehreren Mannschaften bestehen kann, kann er an mehreren Ligen teilnehmen
7. Für einen Verein können beliebig viele Spieler spielen.
8. Jeder Liga ist höchstens eine Aufstiegsliga übergeordnet, in die eine Mannschaft aufsteigen kann.
9. Jeder Liga können bis zu vier Abstiegsligen untergeordnet sein, in die eine Mannschaft absteigen kann.
10. Ein Stadion steht in genau einer Stadt.

Erstellen Sie mit den obigen Entity- und Relationship-Typen ein ER-Diagramm und geben Sie für die Beziehungen zwischen den Entity-Typen den Komplexitätsgrad in der (min, max)-Notation an. Falls ein Entity-Typ an einer Relation mehr als einmal beteiligt ist, machen Sie kenntlich, welcher Entity-Typ welche Rolle spielt:

Aufgabe 3 (7 Punkte)

Das ER-Diagramm aus der letzten Aufgabe sei in folgendes relationale Schema überführt worden:

Städte: {[PLZ, Name]}

Stadien: {[Stadionname, Sitzplätze, Stehplätze]}

Vereine: {[Vereinsname, Anzahl_Titel, Vereinsmitglieder]}

Ligen: {[Liganame, Anzahl_Mannschaften]}

Spieler: {[Name, Tore, PassNr]}

liegen_in: {[Name, PLZ]}

spielen_in: {[Vereinsname, Stadionname]}

nehmen_teil: {[Vereinsname, Liganame]}

spielen_für: {[PassNr, Vereinsname]}

sind_untergeordnet: {[Aufstiegsliga, Abstiegsliga]}

Verfeinern Sie das relationale Schema soweit wie möglich und schreiben Sie es erneut auf.

Aufgabe 4 (8 Punkte)

Gegeben sei das unverfeinerte Schema aus Aufgabe 3:

Städte:	{[<u>PLZ</u> , Name]}
Stadien:	{[<u>Stadionname</u> , Sitzplätze, Stehplätze]}
Vereine:	{[<u>Vereinsname</u> , Anzahl_Titel, Vereinsmitglieder]}
Ligen:	{[<u>Liganame</u> , Anzahl_Mannschaften]}
Spieler:	{[Name, Tore, <u>PassNr</u>]}
liegen_in:	{[<u>Name</u> , PLZ]}
spielen_in:	{[<u>Vereinsname</u> , Stadionname]}
nehmen_teil:	{[<u>Vereinsname</u> , <u>Liganame</u>]}
spielen_für:	{[<u>PassNr</u> , Vereinsname]}
sind_untergeordnet:	{[Aufstiegsliga, <u>Abstiegsliga</u>]}

Formulieren Sie die folgenden Abfragen in der Relationenalgebra:

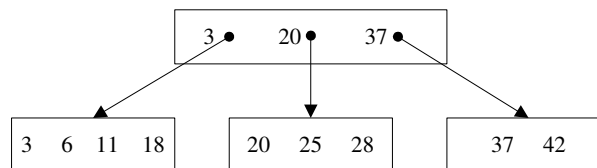
a) Wie heißen die Spieler, die bei dem Verein 'VfL Osnabrück' spielen? (2 Punkte)

b) Welche Vereine spielen nicht in der '1.Bundesliga'? (3 Punkte)

c) In welchen Ligen spielen mindestens alle die Vereine, die im 'Westfalenstadion' spielen? (3 Punkte)

Aufgabe 5 (8 Punkte)

- Zeichnen Sie den folgenden B*-Baum ($k=2$) jeweils
- nach dem Einfügen von 15, 13 und 30 (3 Punkte)
 - nach dem Einfügen von 23 (2 Punkte)
 - nach dem Löschen von 42 (3 Punkte)
- einmal neu.



a) Nach dem Einfügen von 15, 13, 30:

b) Nach dem Einfügen von 23:

c) Nach dem Löschen von 42:

Aufgabe 6 (6 Punkte)

Sei ein B*-Baum gegeben mit $k = 10$ und drei Ebenen.

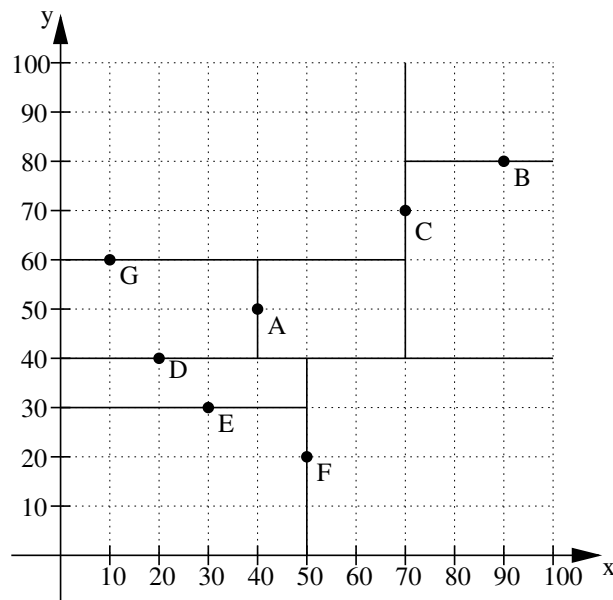
- a) Wieviele Elemente kann der Baum maximal enthalten? (1 Punkt)

- b) Wieviele Elemente kann der B*-Baum minimal enthalten, wenn bislang in den Baum nur Elemente eingefügt wurden, d.h. wenn noch keine Löschoptionen durchgeführt wurden? (3 Punkte)

- c) Wieviele Elemente kann der B*-Baum minimal enthalten, wenn bereits Löschoptionen durchgeführt wurden? (2 Punkte)

Aufgabe 7 (8 Punkte)

Gegeben sei folgende durch die Punkte $A = (40, 50)$, $B = (90, 80)$, $C = (70, 70)$, $D = (20, 40)$, $E = (30, 30)$, $F = (50, 20)$ und $G = (10, 60)$ partitionierte Datenfläche:



- a) Zeichnen Sie den zugehörigen 2-d-Baum mit den jeweiligen Buchstaben und Schlüsseln in den Knoten und den Diskriminierungsbedingungen. (6 Punkte)

- b) Ist die Partitionierung durch die Reihenfolge, in der die Datenrecords in den 2-d-Baum eingefügt wurden, eindeutig festgelegt? (2 Punkte)

Nein

Ja

Aufgabe 8 (11 Punkte)

Gegeben seien die folgenden Relationen einer Universitätsdatenbank:

Professor: {[PersNr, Name, FBnr, Raum]}
Fachbereich: {[FBnr, FBName, DekanPersNr]}
Student: {[MatrNr, Name, Semester, FBnr]}
Pruefung: {[MatrNr, Fach, PrueferPersNr, Note]}

Anm.: die Attribute `DekanPersNr` und `PrueferPersNr` referenzieren einen Professor.
Formulieren Sie folgende Anfragen in SQL:

- a) Wie lauten die Matrikelnummern der Studenten, die eine Prüfung beim Dekan ihres Fachbereichs abgelegt haben? (2 Punkte)

- b) Berechnen Sie die Durchschnittsnote für jeden Studenten. (2 Punkte)

c) Wie lauten die Namen und Matrikelnummern der Studenten mit mindestens vier Semestern, die noch keine Prüfung abgelegt haben? (3 Punkte)

d) Wie lauten die Namen der Studenten, die ausschließlich Prüfungen bei Professoren ihres Fachbereichs abgelegt haben? (4 Punkte)

Aufgabe 9 (6 Punkte)

Gegeben seien die folgenden Relationen aus einer Oldtimerclub-Datenbank (Schlüssel unterstrichen):

Mitglieder: {[Mitgliedsnr, Name, Alter]}
besitzen: {[Mitgliedsnr, Kennzeichen]}
Oldtimer: {[Kennzeichen, Alter, Wert, Farbe]}

Wie lauten folgende SQL-Anfragen umgangssprachlich?

a) SQL-Anfrage:

```
SELECT DISTINCT m.Name
  FROM Mitglieder m, besitzen b, Oldtimer o
 WHERE m.Mitgliedsnr = b.Mitgliedsnr
       AND b.Kennzeichen = o.Kennzeichen
       AND m.Alter < o.Alter
```

Umgangssprachlich (3 Punkte):

b) SQL-Anfrage:

```
SELECT m.Alter
  FROM Mitglieder m, besitzen b, Oldtimer o
 WHERE m.Mitgliedsnr = b.Mitgliedsnr
       AND b.Kennzeichen = o.Kennzeichen
       AND o.Wert = (SELECT max(Wert) FROM Oldtimer)
```

Umgangssprachlich (3 Punkte):

Aufgabe 10 (4 Punkte)

Gegeben sei folgende XML-Datei, die eine Einkaufsliste repräsentiert:

```
<?xml version="1.0" ?>
<!DOCTYPE einkauf SYSTEM "einkaufsliste.dtd">
<einkauf>
  <artikel>
    <beschreibung>Milch</beschreibung>
    <menge>3</menge>
    <einzelpreis>0,55</einzelpreis>
    <gesamtpreis>1,65</gesamtpreis>
  </artikel>
  <artikel>
    <menge>8</menge>
    <einzelpreis>0,99</einzelpreis>
    <gesamtpreis>7,92</gesamtpreis>
  </artikel>
  <artikel>
    <beschreibung>Broetchen</beschreibung>
    <menge>10</menge>
    <einzelpreis>0,25</einzelpreis>
    <gesamtpreis>2,50</gesamtpreis>
  </artikel>
  <artikel>
    <beschreibung>Kiste Wasser</beschreibung>
    <einzelpreis>5,50</einzelpreis>
    <gesamtpreis>5,50</gesamtpreis>
  </artikel>
  <gesamtpreis>17,57</gesamtpreis>
</einkauf>
```

Schreiben Sie für diese XML-Datei eine DTD, gegen die die XML-Datei erfolgreich validiert werden kann. Schränken Sie dabei die Anzahl der Unterelemente sinnvoll ein:

Aufgabe 12 (6 Punkte)

Gegeben sei das Relationenschema

$$R = \{A, B, C, D, E, F\}$$

mit den funktionalen Abhängigkeiten

$$\begin{aligned} BC &\rightarrow C \\ C &\rightarrow AD \\ D &\rightarrow CE \\ E &\rightarrow BC \\ F &\rightarrow D \end{aligned}$$

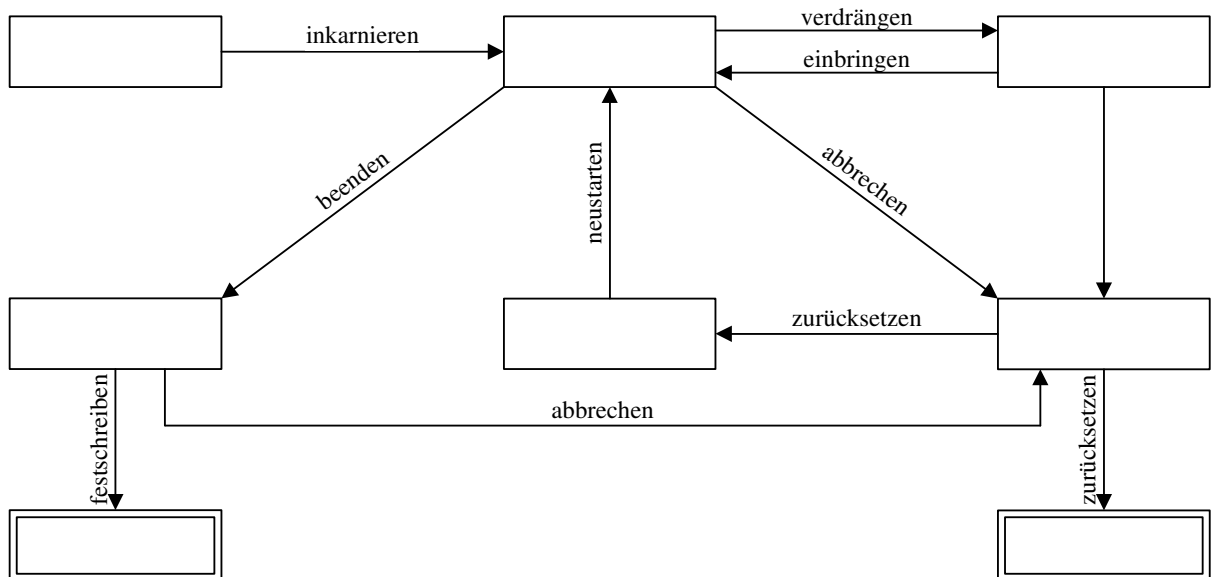
a) Nennen Sie alle Schlüsselkandidaten für R . (2 Punkte)

b) R ist in der 1. Normalform. Ist R auch in der 2. Normalform? Begründen Sie Ihre Antwort. (2 Punkte)

c) Warum ist R nicht in der 3. Normalform? (2 Punkte)

Aufgabe 13 (4 Punkte)

Das folgende Diagramm enthält die Zustandsübergänge einer Transaktion. Benennen Sie die möglichen Zustände:



Aufgabe 14 (6 Punkte)

Betrachten Sie die drei Schedules mit den Transaktionen T_1, T_2, T_3 :

Schedule 1	Schedule 2	Schedule 3
T_1 : BOT	T_1 : BOT	T_1 : BOT
T_2 : BOT	T_2 : BOT	T_2 : BOT
T_3 : BOT	T_3 : BOT	T_3 : BOT
T_2 : lockX(a)	T_2 : lockX(b)	T_1 : lockX(c)
T_3 : lockX(c)	T_3 : lockX(b)	T_2 : lockX(a)
T_3 : lockX(a)	T_1 : lockX(a)	T_3 : lockX(b)
T_1 : lockX(b)	T_1 : lockX(b)	T_2 : unlockX(a)
T_2 : lockX(b)	T_2 : unlockX(b)	T_2 : lockX(c)
T_1 : lockX(c)	T_3 : lockX(a)	T_1 : lockX(b)
...

Zeichnen Sie den Wartegraphen zur Situation am Ende eines jeden Schedules. Liegt zu diesem Zeitpunkt ein Deadlock vor?

Schedule 1: Wartegraph:

Deadlock? Ja Nein

Schedule 2: Wartegraph:

Deadlock? Ja Nein

Schedule 3: Wartegraph:

Deadlock? Ja Nein