

Hauptdiplomklausur Datenbanksysteme I Wintersemester 2001/2002

Name:
Vorname:
Matrikelnummer:
Studienfach:

Wichtige Hinweise:

1. Prüfen Sie Ihr Klausurexemplar auf Vollständigkeit (16 Seiten).
2. Es sind keine Hilfsmittel zugelassen.
3. Die Klausur dauert 100 Minuten.
4. Jede Aufgabe ist auf dem zugehörigen Aufgabenblatt (und ggf. auf separaten Lösungsblättern) zu bearbeiten.
5. Vermerken Sie Ihren Namen und Ihre Matrikelnummer auf jedem Aufgaben- (bzw. Lösungsblatt). Blätter ohne Namens- und Matrikelnummerangabe werden nicht bewertet.
6. Das Deckblatt sowie alle Aufgabenblätter (evtl. Lösungsblätter) sind abzugeben.

	maximale Anzahl Punkte	erreichte Anzahl Punkte
Aufgabe 1	3	
Aufgabe 2	12	
Aufgabe 3	9	
Aufgabe 4	30	
Aufgabe 5	8	
Aufgabe 6	8	
Aufgabe 7	15	
Aufgabe 8	15	
	100	

1. (je 1 Punkt)

Geben Sie für folgende Aussagen an, ob sie wahr oder falsch sind.

Bei einer falschen Antwort werden Punkte abgezogen, die Gesamtpunktzahl kann jedoch nicht unter 0 Punkte sinken.

	wahr	falsch
Ein strenger 2PL-Scheduler liefert immer strikte, serialisierbare Historien	<input type="checkbox"/>	<input type="checkbox"/>
Referentielle Integrität wird in SQL durch foreign key Constraints durchgesetzt	<input type="checkbox"/>	<input type="checkbox"/>
Beim Quorum-Consensus-Verfahren zur Synchronisation von replizierten Daten ist die Summe des Lesequorums und des Schreibquorums immer kleiner als die Gesamtsumme der Gewichte (bzw. Stimmen).	<input type="checkbox"/>	<input type="checkbox"/>

2. (a) (4 Punkte)

Gegeben sei ein Relationenschema $\mathcal{R}(A, B, C, D, E, F)$ mit folgenden funktionalen Abhängigkeiten $\mathcal{F}_{\mathcal{R}} = \{A \rightarrow CD, B \rightarrow AF, C \rightarrow BE, \}$. In welcher höchsten Normalform befindet sich \mathcal{R} ?

(b) (4 Punkte)

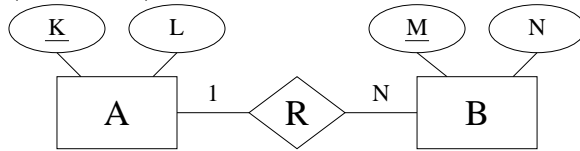
Gegeben sei ein Relationenschema $\mathcal{S}(A, B, C, D, E, F)$ mit folgenden funktionalen Abhängigkeiten $\mathcal{F}_{\mathcal{S}} = \{A \rightarrow B, AE \rightarrow F, BC \rightarrow D, BD \rightarrow E, D \rightarrow C\}$. In welcher höchsten Normalform befindet sich \mathcal{S} ?

(c) (4 Punkte)

Gegeben sei ein Relationenschema $\mathcal{T}(A, B, C, D, E, F)$ mit folgenden funktionalen/mehrwertigen Abhängigkeiten $\mathcal{F}_{\mathcal{T}} = \{A \rightarrow B, B \rightarrow C, C \rightarrow DE, E \rightarrow F, ACD \twoheadrightarrow BEF\}$. In welcher höchsten Normalform befindet sich \mathcal{T} ?

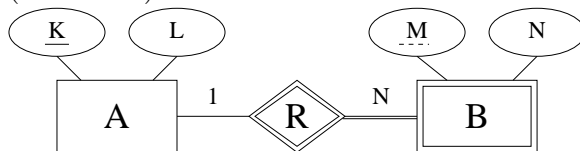
3. In den folgenden Teilaufgaben sehen Sie jeweils einen Ausschnitt aus einem E-R-Diagramm und eine Umsetzung in ein Relationenschema. Leider wurden die Umsetzungen nicht korrekt durchgeführt. Erläutern Sie jeweils kurz, was falsch gemacht wurde und geben Sie eine korrekte Umsetzung der E-R-Diagramme in Relationenschemata an.

(a) (3 Punkte)



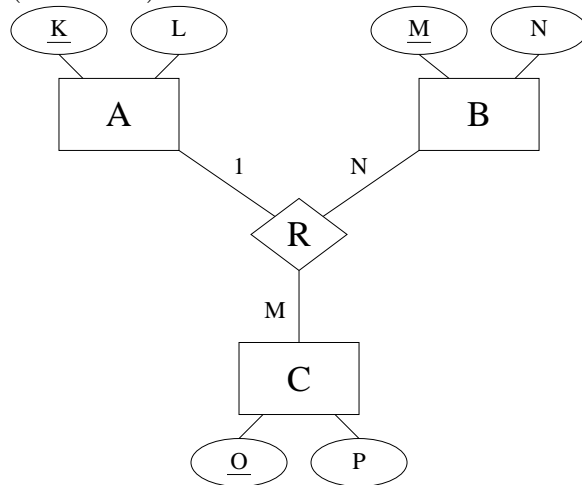
A(K, L, M)
B(M, N)

(b) (3 Punkte)



A(K, L)
B(M, N, K)

(c) (3 Punkte)



A(K, L)
B(M, N, K)
C(O, P, K)

4. Die Fahrtenbücher eines Kurierdiensts werden in einer relationalen Datenbank gehalten. Die Datenbank besitzt folgendes Schema:

- Kurier(PersNr, Name, GebDatum)
- Fahrzeug(Kennzeichen, Typ, kmh)
- Distanz(von, nach, kmZahl)
- Fahrt(FahrtNr, Datum, von, nach, Dauer, Verbrauch, PersNr, Kennzeichen)

Die Relation *Kurier* enthält allgemeine Angaben zu den beschäftigten Kurieren. In der Relation *Fahrzeug* werden das Kennzeichen, der Typ und die Höchstgeschwindigkeit aller Fahrzeuge des Kurierdiensts gespeichert. Die Relation *Distanz* gibt den Abstand zweier Städte in Kilometern an. In der Relation *Fahrt* wird registriert welcher Kurier mit welchem Fahrzeug eine Fahrt zurückgelegt hat. Dabei wird auch die Dauer in Stunden und der Benzinverbrauch in Litern gespeichert.

Formulieren Sie folgende Anfragen in SQL:

(a) (5 Punkte)

Geben Sie die Personalnummern und Namen aller Kuriere an, die am 15.03.2002 von Hamburg nach München unterwegs waren.

Kurier(PersNr, Name, GebDatum)

Fahrzeug(Kennzeichen, Typ, kmh)

Distanz(von, nach, kmZahl)

Fahrt(FahrtNr, Datum, von, nach, Dauer, Verbrauch, PersNr, Kennzeichen)

(b) (5 Punkte)

Geben Sie das Kennzeichen, den Typ und die Höchstgeschwindigkeit des schnellsten Fahrzeugs an.

Kurier(PersNr, Name, GebDatum)

Fahrzeug(Kennzeichen, Typ, kmh)

Distanz(von, nach, kmZahl)

Fahrt(FahrtNr, Datum, von, nach, Dauer, Verbrauch, PersNr, Kennzeichen)

(c) (8 Punkte)

Geben Sie die Personalnummern und Namen der Kurier aus, die bereits mit Fahrzeugen vom Typ „Renault Twingo“ oder „Ford Ka“ unterwegs waren, aber noch nicht mit einem „BMW 750“.

Kurier(PersNr, Name, GebDatum)

Fahrzeug(Kennzeichen, Typ, kmh)

Distanz(von, nach, kmZahl)

Fahrt(FahrtNr, Datum, von, nach, Dauer, Verbrauch, PersNr, Kennzeichen)

(d) (12 Punkte)

Geben Sie die Personalnummer und den Namen des sparsamsten Kuriers aus.
Das ist der Kurier, der den kleinsten Liter pro km Verbrauch aufweist.

5. (8 Punkte)

In einer relationalen Datenbank werden folgende persönliche Daten zu den Angestellten gespeichert.

- Personal(PersNr, Name, GebDatum, Gehalt, PrivatTelNr)

Nicht alle Personen sind autorisiert, alle Daten einzusehen. Eine bestimmte Benutzergruppe bekommt nicht alle Daten zu jeder Person zu sehen, sondern nur eine Auswahl an Attributen. Also z.B.:

PersNr	Name	GebDatum	Gehalt
007	James Bond	12.03.1964	-
203	Gerhard Schröder	07.04.1944	300.000
483	Hans Müller	09.12.1956	-

In der Relation *GehaltSperr*(*PersNr*) sind die Personalnummern der Personen aufgeführt, deren Gehalt ausgeblendet werden soll.

Wie realisieren Sie diese Zugriffsbeschränkung? Beschreiben Sie Ihre konkrete Implementierung.

6. Gegeben ist folgenden SQL-Anfrage auf einer molekularbiologischen Datenbank:

```
select U.Name
from   ChemKlasse C, Unterklasse U, Antibiotikum A,
       ProduziertVon P, Organismus O
where  C.CID = 4
and    C.CID = U.CID
and    U.UID = A.UID
and    A.AID = P.AID
and    P.OID = O.OID
and    O.Name = 'Streptomyces venezuelae';
```

(a) (4 Punkte)

Übersetzen Sie die obige Anfrage in einen kanonischen Operatorbaum.

- (b) (4 Punkte)
Optimieren Sie diesen Operatorbaum nach den in der Vorlesung vorgestellten Regeln.

7. In einem Datenbanksystem wird ein „semi-optimistischer“ Scheduler eingesetzt, d.h. er führt die Operationen nicht sofort aus, sondern hält immer eine Operation zurück. Bei Ankunft der nächsten Operation entscheidet er, welche der beiden Operationen als nächstes ausgeführt wird. Falls beide ohne Probleme ausführbar sind, entscheidet er sich für diejenige die früher angekommen ist. Falls er keine der beiden Operationen konfliktfrei ausführen kann, setzt er entsprechend Transaktionen zurück.

Beispiel: beim Scheduler kommen die Operation in folgender Reihenfolge an:

$w_1[x]$ $r_2[x]$ $w_1[x]$ c_1 c_2

Zuerst kommt $w_1[x]$ beim Scheduler an und wird zurückgehalten. Bei Ankunft von $r_2[x]$ muß er entscheiden, welche Operation er ausführt. Da keine der Ausführungen zu einem Konflikt führt, entscheidet er sich für $w_1[x]$ und hält $r_2[x]$ zurück. Bei Ankunft des nächsten $w_1[x]$ würde das Ausführen von $r_2[x]$ zu einem nicht serialisierbaren Schedule führen, deswegen schickt der Scheduler $w_1[x]$ zur Ausführung (und hält $r_2[x]$ abermals zurück). Bei der Wahl zwischen c_1 , das als nächste Operation ankommt, und $r_2[x]$, wird c_1 losgeschickt, um Probleme mit kaskadierendem Rücksetzen zu vermeiden. Bei $r_2[x]$ und c_2 hat der Scheduler keine Wahl, da die Reihenfolge der Operationen innerhalb einer Transaktion nicht vertauscht werden dürfen und führt deswegen zuerst $r_2[x]$ und danach c_2 aus.

Geben Sie für die folgenden Teilaufgaben jeweils an, ob ein „semi-optimistischer“ Scheduler aus der gegebenen Ankunftsreihenfolge einen Schedule mit den angegebenen Eigenschaften erzeugen kann. Falls ja, geben Sie den Schedule an. Falls nein, begründen Sie Ihre Antwort kurz.

- (a) (5 Punkte)

$w_1[x]$ $r_2[x]$ $r_1[y]$ $w_2[y]$ $r_1[y]$ $w_1[z]$ c_2 c_1

Kann der Scheduler einen rücksetzbaren, serialisierbaren Schedule erzeugen, ohne Transaktionen zurücksetzen zu müssen?

(b) (5 Punkte)

$r_1[x] w_1[y] w_2[y] w_3[y] r_4[y] c_1 c_2 c_3 w_4[z] c_4$

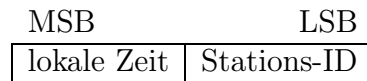
Kann der Scheduler einen serialisierbaren Schedule erzeugen, der kaskadieren-
des Rücksetzen vermeidet, ohne Transaktionen zurücksetzen zu müssen?

(c) (5 Punkte)

$r_1[z] w_1[y] w_1[x] w_2[x] r_3[y] c_1 w_3[z] c_2 c_3$

Kann der Scheduler einen strikten, serialisierbaren Schedule erzeugen, ohne
Transaktionen zurücksetzen zu müssen?

8. Bei einer Zeitstempel-basierten Synchronisation in verteilten Datenbanksystemen reichen Zeitstempel mit der lokalen Zeit der jeweiligen Station nicht aus. Im Buch wird erwähnt, daß die gängigste Methode darin besteht, die lokale Zeit zusammen mit einem eindeutigen Stationsidentifikator zu verwenden:

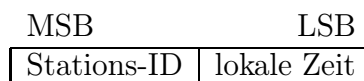


Wobei die Stations-ID die niedrigstwertigen Bits belegt, die lokale Zeit die höchstwertigen.

In den folgenden Teilaufgaben werden weitere Verfahren vorgestellt. Entscheiden Sie zunächst, ob das jeweils angegebene Verfahren funktioniert. Falls nein, geben Sie eine kurze Begründung an. Falls ja, erläutern Sie kurz die Effizienz des Verfahrens im Vergleich zu der obigen, gängigen Methode.

(a) (3 Punkte)

Ein Zeitstempel wird aus der lokalen Zeit mit einem eindeutigen Stationsidentifikator gebildet, jedoch in umgekehrter Reihenfolge:



(b) (3 Punkte)

Ein Zeitstempel besteht aus einem lokalen Zähler, der bei 0 anfängt und für jede Transaktion hochgezählt wird.

(c) (3 Punkte)

Die einzelnen Stationen fordern von einem zentralen Server die Zeitstempel für jede Transaktion an. Der Server generiert für jede Anforderung einen eindeutigen Zeitstempel. Die Werte für die Zeitstempel sind dabei monoton aufsteigend.

(d) (3 Punkte)

Alle Operationen werden auf einem zentralen Server in einer Logdatei mitprotokolliert. Der Zeitstempel einer Operation ist die LSN der Seite auf der die Operation arbeitet.

(e) (3 Punkte)

Ein Zeitstempel besteht aus einem lokalen Zähler, der bei 0 anfängt und für jede Transaktion hochgezählt wird. Dieser Zähler wird noch mit einer eindeutigen Stations-ID versehen:

