

Bitte hier unbedingt
Matrikelnummer und
Adresse eintragen,
sonst keine Bearbei-
tung möglich.

| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|
| | | | | | | | | | |
|--|--|--|--|--|--|--|--|--|--|

Postanschrift: FernUniversität D - 58084 Hagen

Name, Vorname

Straße, Nr.

Plz, Wohnort

FERNUNIVERSITÄT
- Gesamthochschule -
Praktische Informatik I
Universitätstraße 1
D - 58084 Hagen

INF

Fachbereich Informatik

1671 DATENBANKEN I

Klausur

Klausur am 20.03.2004, 10:00 - 13:00 Uhr

- Klausurort:
- | | |
|------------------------------------|----------------------------------|
| <input type="checkbox"/> Berlin | <input type="checkbox"/> Kassel |
| <input type="checkbox"/> Bochum | <input type="checkbox"/> Köln |
| <input type="checkbox"/> Frankfurt | <input type="checkbox"/> München |
| <input type="checkbox"/> Hamburg | <input type="checkbox"/> Bregenz |
| <input type="checkbox"/> Karlsruhe | <input type="checkbox"/> Wien |

Persönliche Angaben:

- Frau Herr geb. am ____ . ____ . 19____
 Vollzeitstudent/Teilzeitstudent Gasthörer

| Aufgabe | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | Summe | Note |
|---------------------|----|----|----|----|----|----|----|---|---|-------|------|
| erreichbar | 14 | 11 | 12 | 19 | 14 | 10 | 20 | | | 100 | |
| erreichte Punktzahl | | | | | | | | | | | |

Hagen, den _____ Korrektor: _____

Klausur zum Kurs 1671 Datenbanken I

Beachten Sie bitte die folgenden Hinweise:

- Die Bearbeitungszeit beträgt 3 Stunden.
- Es sind maximal 100 Punkte erreichbar.
- Als **Hilfsmittel** sind Schreibutensilien und unbeschriebene Papierbögen zugelassen.
- Schreiben Sie mit Tinte oder Kugelschreiber.
- Füllen Sie das **Deckblatt** aus und heften Sie es zusammen mit der **Aufgabenstellung** vor ihre **Lösungsblätter**. Geben Sie alles zusammen nach Beendigung der Klausur ab.
- Schreiben Sie auf jedes Lösungsblatt oben rechts ihren Namen, ihre Matrikelnummer und ggf., zu welcher Aufgabe es gehört.
- Überprüfen Sie bitte die Vollständigkeit der Aufgabenstellungen: Die Klausur umfaßt das Deckblatt, eine Teilnahmebescheinigung, dieses Blatt und 8 Aufgaben.
- Die korrigierte Klausur erhalten Sie in 4 - 8 Wochen zusammen mit einer Musterlösung zurück.
- Sollten Sie eine Klausurteilnahmebescheinigung benötigen, füllen Sie bitte den entsprechenden Vordruck aus und heften Sie ihn hinter das Deckblatt. Die Bescheinigung wird von der FernUniversität abgestempelt und Ihnen zusammen mit der korrigierten Klausur zugeschickt.

Wir wünschen Ihnen viel Erfolg!

Aufgabe 1: 3-Schichten-Architektur eines DBS

(14 Punkte)

Um ein möglichst großes Maß an Datenunabhängigkeit zu erreichen, sollten innerhalb der Architektur eines Datenbanksystems drei verschiedene Datenebenen realisiert werden. Benennen Sie diese Datenebenen und beantworten Sie zu jeder der Ebenen die folgenden zwei Fragen:

- Was beschreibt das jeweilige Modell?
- Von welchen Faktoren hängt die Erstellung des entsprechenden Schemas ab?

Erläutern Sie den Begriff Datenunabhängigkeit anhand der 3-Schichtenarchitektur.

Aufgabe 2: Entity-Relationship-Diagramm

(11 Punkte)

Die Autowerkstatt Car-Repair möchte ihre Einsatzplanung demnächst computergestützt durchführen. Als erster Schritt ist geplant, die notwendigen Auftragsdaten in einer Datenbank zu verwalten. Dazu soll ein Entity Relationship Diagramm als konzeptuelles Schema entwickelt werden.

Dabei sind die folgenden Bedingungen zu beachten:

1. Jeder Kunde besitzt einen Namen und eine Adresse
2. Kunden vergeben Aufträge, die Vergabe der Aufträge erfolgt an einem bestimmten Tag
3. Aufträge besitzen Auftragsbeschreibungen
4. Auftragspositionen enthalten Bearbeitungsdatum, Beschreibung und Arbeitsstunden
5. Jeder Mechaniker besitzt die Attribute Name und Ausbildungsstand
6. Mechaniker bearbeiten Aufträge, ein Auftrag kann von mehreren Mechanikern ausgeführt werden
7. Mechaniker vergeben keine Aufträge
8. Jeder Auftrag besteht aus mindestens einer und maximal beliebig vielen Auftragspositionen. Jede Auftragsposition gehört zu genau einem Auftrag.

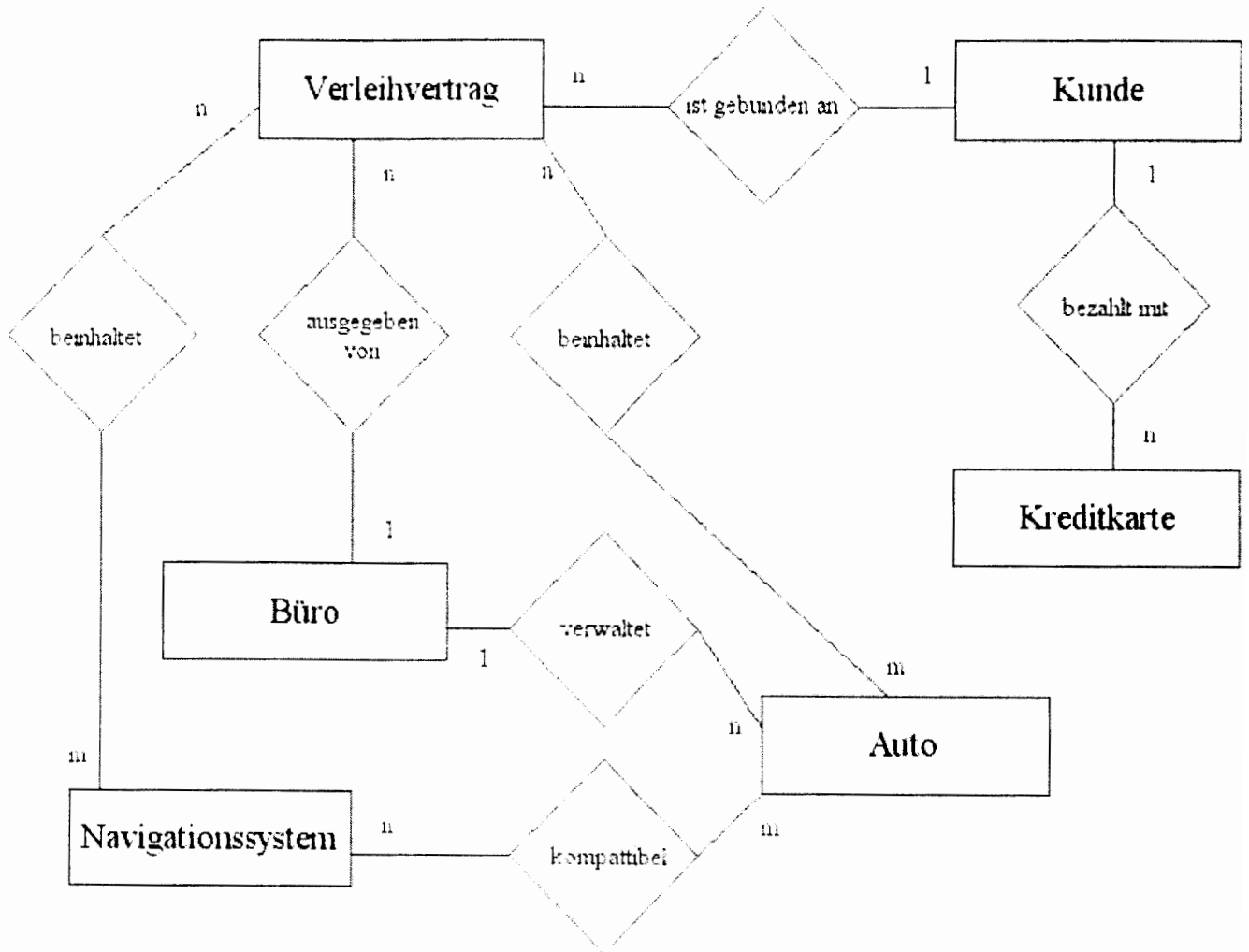
Helfen Sie Car-Repair und erstellen Sie ein Entity Relationship Diagramm, das die Basis-Daten für den geforderten Einsatzzweck beschreibt.

Begründen Sie (kurz!) alle Ihre Entscheidungen (warum Entity, warum Beziehung, warum gerade diese Kardinalität der Beziehung).

Aufgabe 3: Datenmodellierung

(12 Punkte)

Erzeugen Sie aus dem angegebenen ER-Diagramm ein relationales DB-Modell (durch Angabe der Entitäten, der Relationen und beispielhafter Attribute in Textform)!



Aufgabe 4: Queries

19 Punkte

Eine Bibliothek habe zur Verwaltung ihrer Buchausleihe folgende Relationenschemata definiert:

Buch (BuchNr, Autor, Titel, Verlag, Erscheinungsjahr, Standort)

Entleiher (EntleiherNr, Name, Anschrift)

Buch_Ent (BuchNr, EntleiherNr, Entleihdatum)

Formulieren Sie folgende Anfragen

- a) Formulieren Sie in SQL (3 Punkte):
Finde alle Autoren, deren Buch im gleichen Verlag erschienen ist wie das Buch 'Datenbanksysteme' von G. Schlageter.
- b) Formulieren Sie in SQL (4 Punkte):
Finde die Namen und Adressen derjenigen Entleiher, die mehr als 10 Bücher entliehen haben.
- c) Formulieren Sie in der Relationenalgebra (3 Punkte):
Finde die Namen und Adressen aller Entleiher, die nach dem 10.06.2003 ein Buch des Autors G. Schlageter entliehen haben.
- d) Definieren Sie in SQL eine Sicht, die für jeden Verlag die Anzahl der hier erschienenen Bücher ausgibt. (2 Punkte)
- e) Formulieren Sie im Relationenkalkül (3 Punkte):
Finde die Namen und Adressen aller Entleiher, die seit dem 31.12.2003 kein Buch aus dem Addison-Wesley Verlag ausgeliehen haben.
- f) Formulieren Sie im Relationenkalkül (4 Punkte):
Finde die Namen und Adressen aller Entleiher, die alle ihre entliehenen Bücher an einem einzigen Tag ausgeliehen haben.

Aufgabe 5: Funktionale Abhängigkeiten, Normalformen

(14 Punkte)

Geben Sie die aus dem Kurs bekannten Definitionen für

- erste Normalform
- zweite Normalform
- dritte Normalform
- Boyce-Codd Normalform (BCNF):

Gegeben sei die Relation Bestellung mit den Attributen Lief_Nr, Lief_Name, Teile_Nr und Menge sowie den folgenden funktionalen Abhängigkeiten:

Lief_Nr \rightarrow Lief_Name

Lief_Nr, Teile_Nr \rightarrow Menge

Lief_Name, Teile_Nr \rightarrow Menge

- Geben Sie alle Schlüssel der Relation Bestellung an.
- Ist die Relation Bestellung in 3NF?
- Ist die Relation Bestellung in BCNF?

Begründen Sie jeweils Ihre Antwort.

Aufgabe 6: Anfrageoptimierung

10 Punkte

Betrachtet sei die Datenbank einer Leihwagenfirma mit den folgenden Relationen:

Auto (Typ, Hersteller, StammfilialeNr, Kennzeichen)

Filiale (StammfilialeNr, FilialeStraße, FilialeStadt)

Kunde (Name, Straße, Wohnort, AusweisNr)

Vertrag (AusweisNr, Kennzeichen, Datum)

An diese Datenbank wird folgende Anfrage gestellt:

Vertrag [AusweisNr = AusweisNr] Kunde [Kennzeichen = Kennzeichen] Auto [Datum < 12.03.2004] [Wohnort = 'Hagen'] [Straße = 'Universitätsstr. 1'] [Name = 'Bratbäcker' AND Hersteller = 'VW'] [Datum, Typ, Hersteller, Name, Wohnort]

- Zeichnen Sie für den obigen Ausdruck den entsprechenden Operatorbaum. (2 Punkte)
- Optimieren Sie den angegebenen Ausdruck mit Hilfe der Ihnen bekannten Regeln (begründen Sie!) und geben Sie den optimierten Ausdruck an (6 Punkte).
- Zeichnen Sie den zugehörigen Operatorbaum (2 Punkte).

Aufgabe 7: SQL

20 Punkte

Beantworten Sie bitte **kurz** folgende Fragen:

- a) Was versteht man in Bezug auf die relationale Abfragesprache SQL unter einer *Tupelvariablen* und wann ist deren Verwendung notwendig? (1 Punkt)
- b) Was versteht man unter einem *View*? (1 Punkt)
- c) Was versteht man unter einer *Basisrelation*? Ist ein *View* eine *Basisrelation*? (1 Punkt)
- d) Was versteht man unter *Embedded SQL* und welche Rolle spielt dabei der sog. *Precompiler*? (1 Punkt)

Gegeben seien folgende Relationen (alle Attribute seien vom Typ INT):

r1

| a | b | c |
|-----|-----|-----|
| 20 | 40 | 60 |
| 80 | 100 | 120 |
| 140 | 160 | 160 |
| 20 | 100 | 180 |

r2

| a | b | c | d |
|----|-----|-----|-----|
| 22 | 76 | 60 | 38 |
| 80 | 4 | 120 | 128 |
| 14 | 176 | 196 | 208 |
| 20 | 102 | 58 | 22 |

Welche Ergebnisse liefern die folgenden Abfragen?

- e) `select b, a from r1
where c in
 (select c from r2
 where d <
 (select max(d) from r2));` (3 Punkte)
- f) `select max(c) from r1
where b >
 (select min(c) from r2
 where d > 20)` (2 Punkte)
- g) `select avg(d) from r2`
(gehen Sie - konform zu ANSI SQL - davon aus, daß avg eine Fließkommazahl zurückgibt) (2 Punkte)

Konstruieren Sie bitte SQL-statements für folgende Aufgaben:

- h) Konstruieren Sie eine Abfrage, die alle **unterschiedlichen** Tupel der Relation r2 ausgibt, bei denen **gleichzeitig** der Wert des Attributes d um mehr als 10 größer ist als der eines anderen Tupels der Relation r2 und bei denen der Wert des Attributes a um mehr als 10 größer ist als der eines anderen Tupels der Relation r2. Geben Sie das Ergebnis der Abfrage an. (5 Punkte)

- i) Konstruieren Sie eine Abfrage, die das Attribut b aller Tupel der Relation r_1 ausgibt, deren Werte in jeder Komponente größer sind als der Durchschnitt dieser Komponente über alle Tupel der Relation r_1 . . Geben Sie das Ergebnis der Abfrage an. **(4 Punkte)**