

Musterlösung zur Haupt-Klausur 1671

WS 2005 / 2006

Aufgabe 1: Datenbank, DBMS, DBS

(7 Punkte)

integriert: Die Daten sind entsprechend der natürlichen Zusammenhänge in der Anwendungswelt strukturiert - und nicht danach, wie einzelne Anwendungen die Daten benötigen. (0,5 Punkte)

gemeinsame Basis: Die Daten in der Datenbank können durch viele Benutzer genutzt werden, wobei sie dann unterschiedliche Sichten auf diese Daten haben werden. (0,5 Punkte)

Datenbankmanagementsystem: Das Datenbankmanagementsystem (DBMS) ist ein Softwaresystem, das es ermöglicht, eine Datenbank zu definieren, Daten zu speichern, zu verändern und zu löschen, so wie Anfragen an die Datenbank zu stellen. (1 Punkt)

Datenbanksystem: Datenbank und Datenbankmanagementsystem (DBMS) bilden zusammen ein Datenbanksystem. (1 Punkt)

Datenunabhängigkeit: (insgesamt 3 Punkte)

Datenunabhängigkeit bedeutet, dass Anwendungsprogramme von Änderungen auf der internen und der konzeptuellen Ebene unberührt bleiben. Man unterscheidet physische und logische Datenunabhängigkeit. (1Punkt)

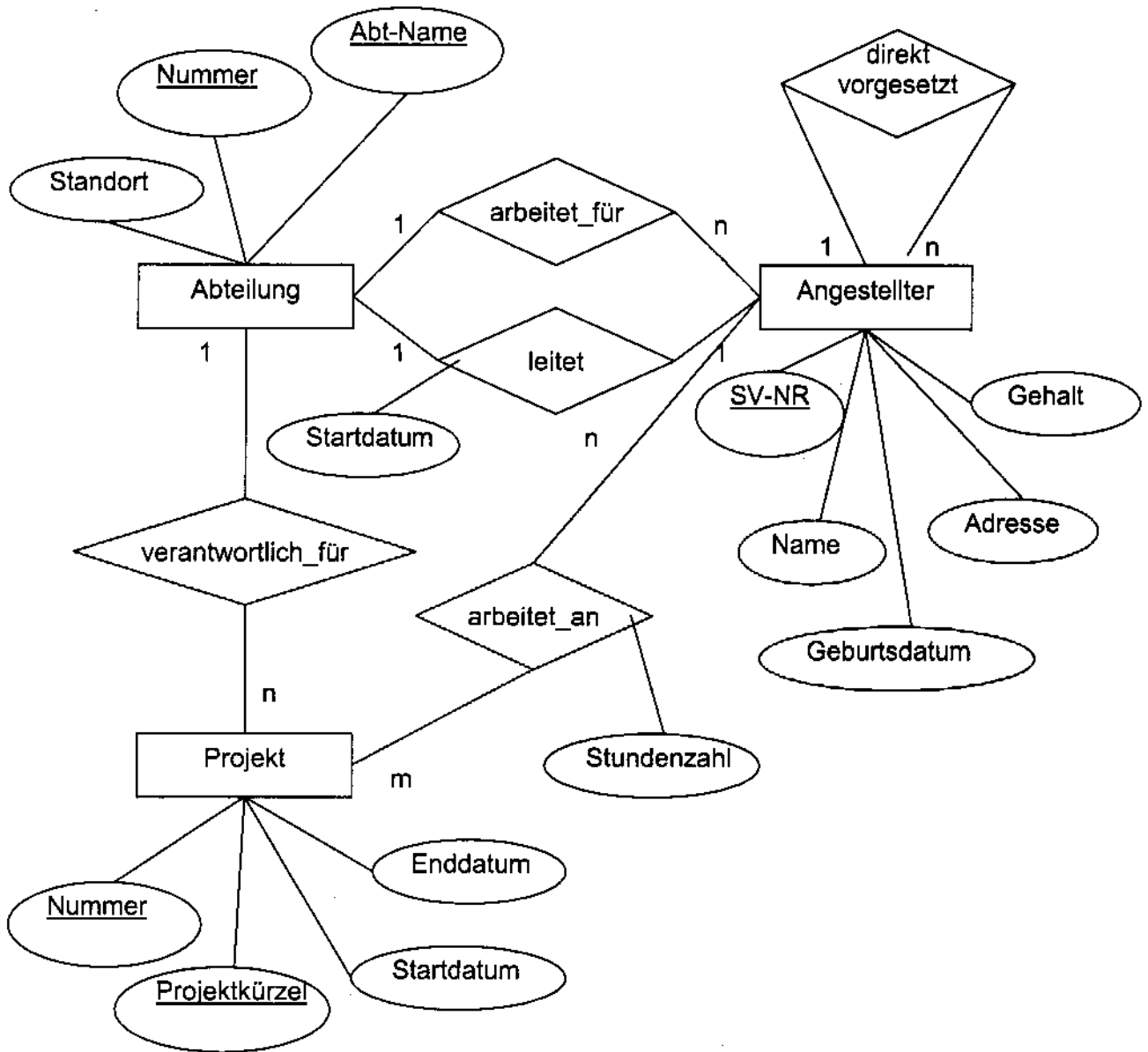
Physische Datenunabhängigkeit bedeutet Isolierung der Anwendungsprogramme vor Änderungen der physischen Datenorganisation. (1Punkt)

Logische Datenunabhängigkeit bedeutet Isolierung der Anwendungsprogramme vor Änderungen des konzeptuellen Modells. (1Punkt)

Integrität der Daten: Korrektheit und Vollständigkeit der abgespeicherten Daten (-beides zu messen an der Realität, über die die Datenbank Daten enthält). (1 Punkt)

Aufgabe 2: ER-Relationship Modellierung

(16 Punkte)



SV-NR: Sozialversicherungsnummer
Abt-Name: Abteilungsname

Aufgabe 3: Formulierung von Queries

(15 Punkte)

a) (3 Punkte)

```
SELECT      AUTOR
FROM        BUCH
WHERE       VERLAG =
            (SELECT  VERLAG
             FROM    BUCH
             WHERE   AUTOR = 'R. Ramakrishnan'
              AND    TITEL = 'Database Managementsystems')
```

b) (3 Punkte)

```
BUCH [AUTOR = 'A. Silberschatz'] [B_NR = B_NR] BUCH_ENT
[ENTLEIHDATUM > 18.03.2005] [E_NR = E_NR] ENTLEIHER [NAME,
ANSCHRIFT]
```

c) (2 Punkte)

```
DEFINE VIEW ZAHL_DER_BÜCHER (STANDORT, ANZAHL) AS
SELECT  STANDORT, COUNT (B_NR)
FROM    BUCH
GROUP BY STANDORT
```

d) (3 Punkte)

Diese Aufgabe war missverständlich formuliert, man kann sie, je nach dem wie man sie liest, auf 2 Arten verstehen:

(1) Finde die Namen und Anschriften aller Entleiher, die seit dem 15.02.2005 kein Buch aus dem 'W.H. Freeman & Co. Verlag' mehr ausgeliehen haben.

(2) Finde die Namen und Anschriften aller Entleiher, die nach dem 15.02.2005 ein Buch aus dem 'W.H. Freeman & Co. Verlag' ausgeliehen haben.

Beide folgenden (bzw. sinngleichen) Lösungen werden als korrekt bewertet.

(1)

```
SELECT NAME, ANSCHRIFT
FROM    ENTLEIHER E
WHERE   NOT EXISTS
        ( SELECT  BE.E_NR
          FROM    BUCH_ENT BE, BUCH B
          WHERE   BE.ENTLEIHDATUM > 15.02.2005
            AND   BE.E_NR = E.E_NR
            AND   BE.B_NR = B.B_NR
            AND   B.VERLAG = 'W.H. Freeman & Co. Verlag')
```

```

(2)
SELECT      NAME, ANSCHRIFT
FROM        ENTLEIHER
WHERE       E_NR IN
           ( SELECT  E_NR
             FROM    BUCH_ENT
             WHERE   ENTLEIHDATUM > 15.02.2005
             AND     B_NR IN
                   ( SELECT B_NR
                     FROM BUCH
                     WHERE VERLAG = 'W.H. Freeman & Co. Verlag'
                   )
           )

```

e) (4Punkte)

```

RANGE BUCH_ENT X
RANGE BUCH_ENT Y
{ENTLEIHER.NAME, ENTLEIHER.ANSCHRIFT | ∃ X (X.ENR =
ENTLEIHER.E_NR ∧ ∃ Y (Y.E_NR = X.E_NR ∧ Y.ENTLEIHDATUM <>
X.ENTLEIHDATUM))}

```

Aufgabe 4: Normalformen

(14 Punkte)

a) (5 Punkte)

Die Relation Einkaufs_Abt ist in 2NF, da die Nichtschlüsselattribute Produkt_Gruppe, EK_Lager und Lager_Leiter voll funktional abhängig sind vom Schlüssel Abt_Name. Relationen mit einstelligem Schlüssel sind immer in 2NF.

Einkaufs_Abt ist nicht in 3NF, da es transitive Abhängigkeiten gibt. Genauer: Es gibt den Schlüssel Abt_Name, einen Nichtschlüssel Produkt_Gruppe und ein Nichtschlüsselattribut EK_Lager, und es gilt

Abt_Name	-> Produkt_Gruppe	
Produkt_Gruppe	-> EK_Lager, aber	
EK_Lager	-> Abt_Name	gilt nicht.

Damit ist Einkaufs_Abt auch nicht in BCNF. (3 Punkte)

Die Relation Bestellung ist nicht in 2NF, da die Nichtschlüsselattribute L_Anschrift und Produkt_Gruppe nicht voll funktional abhängig sind vom Schlüssel (Lieferanten_Bez, Waren_Bez, Best_Datum). Damit ist Bestellung auch nicht in 3NF und somit auch nicht in BCNF. (2 Punkte)

b) (4 Punkte)

Einfüge-Anomalie: In der Relation BESTELLUNG kann kein neuer Lieferant mit Anschrift gespeichert werden, bei dem nicht zumindest eine Ware bestellt wurde. (2 Punkte)

Andere Beispiele: Eine Produktgruppe und seine Zuordnung zu einem Einkaufslager kann erst gespeichert werden, wenn eine Abteilung dafür zuständig ist. Der Lagerleiter eines Lagers kann erst gespeichert werden, wenn mindestens eine Abteilung existiert, die eine Produktgruppe in diesem Lager einkauft.

Lösch-Anomalie: Wird die derzeit einzige Bestellung bei einem Lieferanten storniert, so gehen mit dem Löschen dieses Datensatzes alle Informationen über diesen Lieferanten verloren. (1,5 Punkte) (Ähnliche Beispiele gibt es auch für die Tabelle Einkaufsabt.)

Änderungs- Anomalie: Ändern sich die Adresse eines Lieferanten, so muss in allen Einträgen von Bestellungen, die diesen Lieferanten enthalten, diese Änderung durchgeführt werden. Entspr: wird ein neuer Lagerleiter für ein Lager eingesetzt, so muss in allen Tupeln der Relation Einkaufs_Abt, in denen eine Produktgruppe aus diesem Lager vorkommt, diese Änderung durchgeführt werden (1,5 Punkte)

c) (4 Punkte)

Einkaufs_Abt (Abt_Name, Abt_Leiter, Produkt_Gruppe, EK_Lager, Lager_Leiter)

→

Einkaufs_Abt (Abt_Name, Abt_Leiter, Produkt_Gruppe)

Lagerzuordnung (Produkt_Gruppe, EK_Lager)

Lager (EK_Lager, Lager_Leiter)

Bestellung (Lieferanten_Bez, L_Anschrift, Waren_Bez, , Produkt_Gruppe, Menge, Best_Datum)

→

Bestellung (Lieferanten_Bez, Waren_Bez, Menge, Best_Datum)

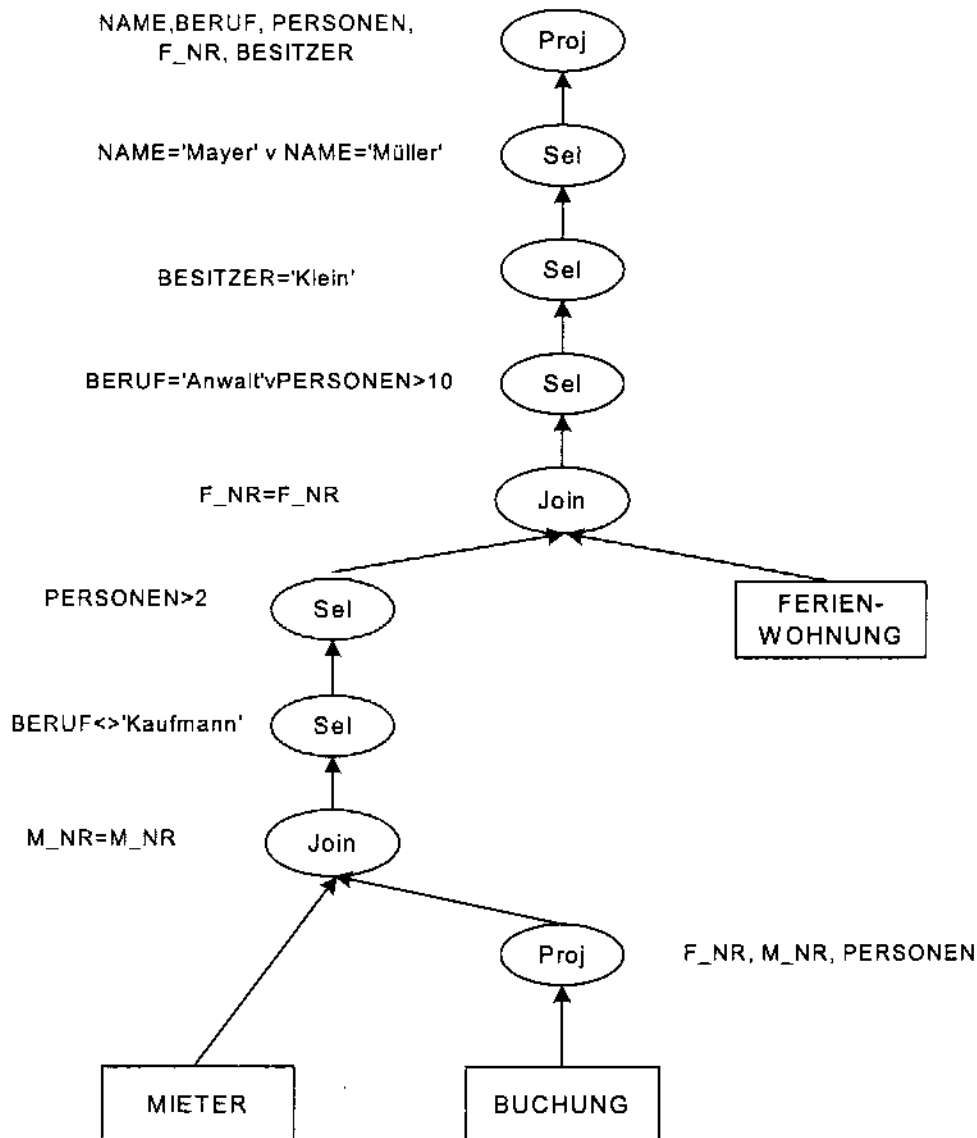
Lieferant (Lieferanten_Bez, L_Anschrift)

Produkt_Klassifizierung (Waren_Bez, Produkt_Gruppe)

Aufgabe 5: Query-Optimierung

(14 Punkte)

(a) Operatorbaum (2 Punkte)



(b) (6 Punkte) Die Selektion [PERSONEN > 2] bezieht sich nur auf die Relation BUCHUNG und kann daher vor den Verbund [M_NR = M_NR] und ebenfalls vor der Projektion [F_NR, M_NR, PERSONEN] ausgeführt werden.

[NAME = ,Mayer' v NAME = ,Müller'] und [BERUF <> ,Kaufmann'] sind Selektionen, die sich beide auf die Relation MIETER beziehen. Deshalb ist es möglich, diese zu einer Selektion [BERUF = ,Kaufmann' und (NAME = ,Mayer' v NAME = ,Müller')] zusammenzufassen und sie vor der Verbundoperation [M_NR = M_NR] auszuführen.

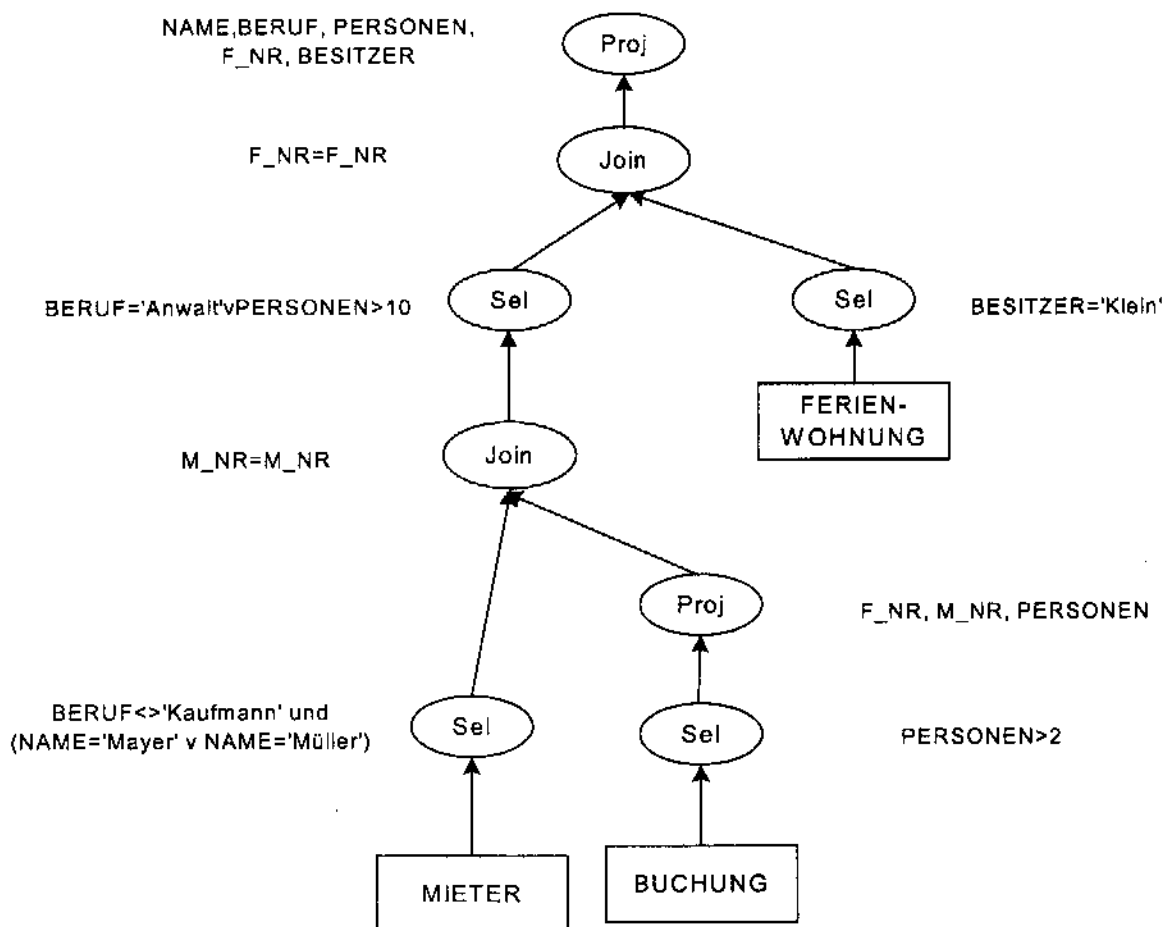
Die Selektion [BESITZER = ,Klein'] bezieht sich nur auf die Relation FERIENWOHNUNG und kann daher vor dem Verbund [F_NR = F_NR] ausgeführt werden.

Die Selektion [BERUF = ‚Anwalt‘ v PERSONEN > 10] kann, da es sich hier um eine ODER-Verknüpfung handelt, nicht aufgespalten werden. Sie bezieht sich jedoch nur auf die beiden Relationen MIETER und BUCHUNG und kann daher auch vor dem Verbund [F_NR = F_NR] ausgeführt werden.

Als Resultat der Optimierung ergibt sich damit der folgende Ausdruck:

MIETER [BERUF <> ‚Kaufmann‘ und (NAME = ‚Mayer‘ v NAME = ‚Müller‘)]
 [M_NR = M_NR] (BUCHUNG [PERSONEN > 2] [F_NR, M_NR, PERSONEN])
 [BERUF = ‚Anwalt‘ v PERSONEN > 10] [F_NR = F_NR] (FERIENWOHNUNG
 [BESITZER = ‚Klein‘]) [NAME, BERUF, PERSONEN, F_NR, BESITZER]

Operatorbaum: (2 Punkte)



(c) Die notwendigen Verbund-Operationen (z.B. R [A = B] S) lassen sich auf physischer Ebene noch optimieren durch:

- (2 Punkte) Ausnutzung von Sekundärindexen:

Existiert ein Index für B, dann ist für jedes Tupel in R sofort die zugehörige Tupelmenge in S feststellbar – und damit die Tupelmenge des Verbundes konstruierbar. Die Zahl der notwendigen Zugriffe auf Tupel wird damit linear zur Größe der beiden Relationen.

- (2 Punkte) Sortieren von R nach A und von S nach B:
Bei effizienten Sortierverfahren werden $O(n \log n)$ Schritte benötigt. Das anschließende Erstellen des Verbundes ist dann sehr schnell: R und S werden parallel durchlaufen, wobei jeweils auf gleiche Werte für A und B zu prüfen ist.

Aufgabe 6: Umwandlung eines ER-Modells in ein relationales Schema (9 Punkte)

Gärtner (G-Name, Adresse)

Entity Typ wird Relation (1 Punkt)

Pflanze (P-Name, Art, G-Name, S-Nr)

Entity Typ wird Relation, 1:n-Beziehung wird über Fremdschlüssel gebildet (3 Punkte)

Nährstoffe (N-Nr, N-Name, Lieferant)

Entity Typ wird Relation (1 Punkt)

Standort (S-Nr, Größe)

Entity Typ wird Relation (1 Punkt)

Nährstoffbedarf (P-Name, N-Nr, Menge)

n:m Beziehung wird auf Relation abgebildet, Attribut der Beziehung wird Attribut der Relation (2 Punkte)

Lagerung (N-Nr, S-Nr)

n:m Beziehung wird auf Relation abgebildet (1 Punkt)

Aufgabe 7: Daten- und Schemamanipulation in SQL

(11 Punkte)

a) (2 Punkte)

```
INSERT INTO Buch_Ent (23, 47, '5.1.2006')
```

oder:

```
INSERT INTO Buch_Ent  
(B_Nr, E_Nr, Entleihdatum)  
VALUES (23, 47, '5.1.2006')
```

b) (2 Punkte)

```
DELETE FROM Buch_Ent  
WHERE E_Nr = 47  
AND B_Nr = 23
```

c) (2 Punkte)

```
UPDATE Entleiher  
WHERE E_Nr = 47  
SET Anschrift = 'Buchstr. 8a, Buchdorf'
```

d) (4 Punkte)

```
Buch ( B_Nr, Autor, Titel, Verlag, Erscheinungsjahr, StandortNr )  
Entleiher ( E_Nr, Name, Anschrift )  
Buch_Ent ( B_Nr, E_Nr, Entleihdatum )  
Standort ( StandortNr, Standort, Themengebiet, Erstelldatum )
```

e) (1 Punkte)

```
CREATE TABLE Standort  
(StandortNr INTEGER NOT NULL,  
Standort CHAR(40),  
Themengebiet CHAR(40),  
Erstelldatum CHAR(8),  
PRIMARY KEY (StandortNr))
```

Aufgabe 8: Multiple-Choice

(5 Punkte)

a) Normalformen und Schlüssel:

- X ist *Schlüssel* von $\{A_1, \dots, A_n\}$ genau dann, wenn $X \rightarrow \{A_1, \dots, A_n\} \in F^+$ und $\{A_1, \dots, A_n\}$ ist voll funktional abhängig von X.
- Wenn eine Relation in 2. Normalform vorliegt, so enthält sie kein Nichtschlüsselattribut, von dem andere Nichtschlüsselattribute funktional abhängig sind
- Wenn eine Relation in 3NF ist und nur einelementige Schlüsselkandidaten besitzt, so ist sie auch in BCNF (Boyce-Codd Normalform)

b) Um ein möglichst großes Maß an Datenunabhängigkeit zu erreichen, sollten drei verschiedene Datenebenen realisiert werden. Dies sind:

- Attribute, Entities und Beziehungen
- Externes, konzeptuelles und internes Schema
- Datensystem, Zugriffssystem und Speichersystem

c) Eine Anfragesprache wird als relational vollständig bezeichnet,...

- wenn sie die gleiche Sprachmächtigkeit wie das Relationenkalkül besitzt
- wenn sie die gleiche Sprachmächtigkeit wie SQL besitzt
- wenn sie die gleiche Sprachmächtigkeit wie die Relationenalgebra besitzt
- wenn sie die JOIN-Funktion enthält
- wenn sie die booleschen Funktionen enthält und algebraisch optimiert werden kann
- wenn jede berechenbare Funktion berechnet werden kann