

DBIS-Nachklausur Sommersemester 2012
Prof. Dr. Georg Lausen
Dauer: 90 Minuten
Punkte: 90

Dies ist ein Gedankenaufschrieb. Keine Garantie für die Korrektheit aller Aufgaben.

1. Aufgabe: SQL-Anfragen

4+5+6 = 15 Punkte

Gegeben sei folgende Tabelle:

Verbindung(Von, Nach, Dauer, Distanz)

- a) Geben Sie für jede Verbindung die durchschnittliche Geschwindigkeit aus, welche dem Quotienten von (Distanz/Dauer) entspricht.
 - b) Geben Sie alle Verbindungen an, deren Dauer der maximalen Dauer von allen Verbindungen entspricht.
 - c) Geben Sie für jede Stadt V die Verbindung mit maximaler Dauer an.

2. Aufgabe: Relationenalgebra und Relationenkalkül

4 + 4 + 4 = 12 Punkte

- a) Gegeben sei folgende Tabelle:

| LCode | Organisation |
|-------|-------------------------|
| A | EU |
| A | NATO |
| RU | Random 3rd Organisation |
| RU | Random 4th Organisation |
| D | EU |
| D | NATO |
| F | EU |
| F | NATO |
| F | Random 3rd Organisation |

Was ist das Ergebnis der folgenden Anfrage?

```

SELECT DISTINCT M1.LCode
FROM Mitglied M1, Mitglied M2
WHERE M2.LCode ='A' AND M1.Organisation = M2.Organisation
GROUP BY M1.LCode
HAVING COUNT(M1.Organisation)
= (SELECT COUNT(M3.Organisation)
  FROM Mitglied M3
  WHERE M3.LCode = 'A');

```

- b) Gegeben: Ausdruck in relationaler Algebra. Korrigieren Sie ihn, so dass er äquivalent zu obigem SQL-Statement ist.

$$\sigma[LCode = 'A']Mitglied \div \pi[Organisation]Mitglied$$

- c) Gegeben: Ausdruck im Relationenkalkül. Korrigieren Sie ihn, so dass er äquivalent zu obigem SQL-Statement ist.

$$\{X: LCode \mid \forall Y \text{ Mitglied}(X, Y) \rightarrow \text{Mitglied}('A', Y)\}$$

3. Aufgabe: Funktionale Abhangigkeiten

4 + 4 = 8 Punkte

Gegeben sei $V = \{A, B, C, D\}$ und folgende funktionale Abhangigkeit: $F = \{A \rightarrow B, B \rightarrow C\}$

- a) Geben Sie alle nicht-trivialen Abhangigkeiten an die in F^+ enthalten sind.

b) Geben Sie alle nicht-trivialen Abhangigkeiten an die in F^+ nicht enthalten sind.

4. Aufgabe: Funktionale Abhangigkeiten**(4+4) + 2 = 10 Punkte**

Gegeben ist die folgende Menge der funktionalen Abhangigkeiten:

$$F = \{A \rightarrow B, B \rightarrow C, BC \rightarrow A\}$$

- a) Berechnen Sie F^{min} , fur die Linksreduktion mussen Sie nur $BC \rightarrow A$ betrachten, fur die Rechtsreduktion nur $A \rightarrow B$.

- b) Geben Sie alle Schlessel an und begrunden Sie warum diese Schlessel sind.

5. Aufgabe**2 + 6 = 8 Punkte**

Gegeben sind folgende Relationsschemata: $R(A,)$, $S(C, D)$ mit folgenden Instanzen $R = r$, $S = s$ mit Größe N bzw. M. Die Tupel aus R und S sind gleich groß. Eine Seite kann k Tupel fassen, k ist immer gerade. Es wird berechnet: $R \bowtie_{B=C} S$.

- a) Berechnen Sie bezüglich der Verbundgröße die maximale untere Schranke.
- b) Berechnen Sie bezüglich der Verbundgröße die minimale obere Schranke.

6. Aufgabe: SQL

a) *CREATE TABLE R(*

*A NUMBER NOT NULL,
B NUMBER NOT NULL,
C NUMBER NOT NULL,
PRIMARY KEY (A);*

Fügen Sie eine Assertion-Klausel hinzu sodass folgende Abhängigkeit erfüllt ist: $B \rightarrow C$.

b) *CREATE TABLE S(*

*A NUMBER NOT NULL,
D NUMBER NOT NULL,
C NUMBER NOT NULL,
PRIMARY KEY (A),
FOREIGN KEY (D) REFERENCES R(A));*

Ersetzen Sie den Foreign Key durch eine Check-Klausel sodass die Abhängigkeit bewahrt wird.

7. Aufgabe: Serialisierbarkeit von Transaktionen

2 + 7 = 9 Punkte

Gegeben sind folgende Transaktionen T1, T2, T3.

T1: RA WC

T2: RB WA

T3: RC WB

a) Sind alle möglichen Schedules von T1, T2, T3 serialisierbar?

b) Gegeben ist folgender Schedule.

$$S = R1A \ R2B \ W2C \ R3C \ W3A \ W1B$$

Fügen sie Lock- $L_i X$ und Unlock-Schritte $U_i X$ für $i \in \{1,2,3\}$ und $X \in \{A,B,C\}$ ein, so dass der Schedule S mit dem 2PL-Verfahren realisierbar ist.

8. Aufgabe: ER-Diagramm

Gegeben sind folgende drei Relationsschemata: Zeichen Sie das zugehörige ER-Diagramm mit allen Beziehungskardinalitäten. Attribute müssen nicht angegeben werden.

- a) Lieferant(LNr, Name, ...) Bauteil(BNr, Name, ...) PBL(LNr, BNr, PNr) Produkt(PNr, ...)
- b) Lieferant(LNr, Name, ...) Bauteil(BNr, Name, ..., PNr, LNr), Produkt(PNr, Name, ...)
- c) PBL(PNr, BNr, LNr, ...)

9. Aufgabe

Gegeben sind die Relationen $T(A,B)$, $S(B,C)$, $R(C,D)$, $Q(D,E)$, über die ein Verbund erstellt werden soll.

- a) Geben Sie einen Left-Deep-Tree für den Verbund an. Vermeiden Sie kartesische Produkte.

- b) Geben Sie einen Bushy-Tree für den Verbund an. Vermeiden Sie kartesische Produkte.

Datenbanken und Informationssysteme

SS 2009

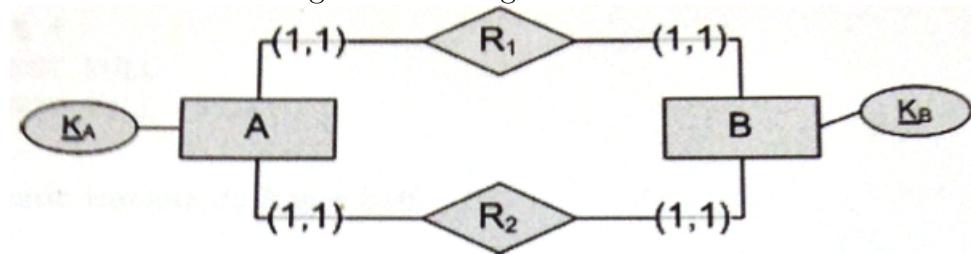
Prof. Dr. G. Lausen

17. September 2009

1 Aufgabe

(8+2 Punkte)

Betrachten Sie das folgende ER-Diagramm:



Die folgende Entitätstabellen:

| A | K_A |
|---|-------|
| | a_1 |
| | a_2 |

| B | K_B |
|---|-------|
| | b_1 |
| | b_2 |

Und die folgenden Varianten für die Beziehungstabellen:

i)

| R_1 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |

| R_2 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |

ii)

| R_1 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |
| | a_2 | b_2 |

| R_2 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |
| | a_2 | b_2 |

iii)

| R_1 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |
| | a_2 | b_2 |

| R_2 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_2 |
| | a_2 | b_1 |

iv)

| R_1 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_1 |
| | a_2 | b_1 |

| R_2 | K_A | K_B |
|-------|-------|-------|
| | a_1 | b_2 |
| | a_1 | b_1 |

a)

Welche der Beziehungstabellen (i)- (iv) erfüllen Beziehungskomplexitäten, welche nicht?
Begründen Sie!

b)

Ändern Sie obige Beziehungskomplexitäten so ab, dass alle Beziehungstabellen (i) - (iv)
sie erfüllen.

2 Aufgabe

(5+5 Punkte)

Sei in SQL eine Tabelle R definiert wie folgt:

```
CREATE TABLE R (
A INTEGER NOT NULL,
B INTEGER NOT NULL)
```

Und sei folgende Instanz zu R gegeben:

| R | A | B |
|---|---|---|
| 1 | 1 | |
| 1 | | 2 |
| 2 | 1 | |
| 2 | | 2 |
| 3 | | 5 |

Betrachten Sie folgende SQL-Anfragen:

(i)

```
SELECT B
FROM R
GROUP BY A
```

(ii)

```
SELECT A
FROM R
GROUP BY A
```

(iii)

```
SELECT count(B)
FROM R
GROUP BY A
```

(iv)

```
SELECT A, count(B)
FROM R
GROUP BY A
```

(v)

```
SELECT max(C)
FROM ( SELECT count(B) AS C
FROM R GROUP BY A ) T
```

a)

Welche der SQL-Ausdrücke (i)-(v) sind syntaktisch fehlerhaft? Begründen Sie!

b)

Geben Sie für alle syntaktisch korrekten Ausdrücke das Ergebnis an.

3 Aufgabe

(15 Punkte)

Sei in SQL eine Tabelle K definiert wie folgt:

```
CREATE TABLE K (
    X INTEGER,
    Y INTEGET )
```

a)

Schreiben Sie eine SQL-Anfrage, die angewendet auf eine Instanz zu K als Ausgabe eine Ergebnismenge von Tupeln der Form (k, m, n) genau dann erzeugt, wenn die betrachtete Instanz zu K Tupel (k, m) und (m,n) enthält.

b)

Erweitern Sie ihre Anfrage aus a) so, dass für jedes Ergebnistupel (k, m, n) gerade gilt:
 $k < m < n$

c)

Welche Ergebnisse liefern die Anfragen aus a) und b) wenn angewendet auf die Instanz:

| K | X | Y |
|------|------|---|
| 1 | 2 | |
| 2 | 3 | |
| 3 | 1 | |
| 1 | null | |
| null | null | |

4 Aufgabe

(6 + 4 Punkte)

Sei ein Relationsschema R mit Attributmenge

$$V = \{A, B, C, D, E\}$$

und die Menge von funktionalen Abhangigkeiten

$$\mathcal{F} = \{AB \rightarrow C, BC \rightarrow D, CD \rightarrow E, DE \rightarrow A\}$$

gegeben.

a)

Welche der folgenden funktionalen Abhangigkeiten sind in \mathcal{F}^+ enthalten:

1. $AB \rightarrow D$
2. $AB \rightarrow E$
3. $AB \rightarrow A$
4. $A \rightarrow A$
5. $A \rightarrow B$
6. $A \rightarrow C$

Begrunden Sie jeweils.

b)

Geben Sie alle Schlessel zu R an.

5 Aufgabe

(8 Punkte)

Betrachten Sie Relationen der Form:

| R | A | B | S | B | C | T | A | B |
|-----|----------|----------|-----|----------|----------|-----|----------|----------|
| | 1 | 2 | | 2 | 1 | | 2 | 1 |
| | 2 | 2 | | 2 | 2 | | 2 | 2 |
| | 3 | 2 | | 2 | 3 | | 2 | 3 |
| | \vdots | \vdots | | \vdots | \vdots | | \vdots | \vdots |
| | n | 2 | | 2 | m | | 2 | m |

Geben Sie jeweils an, wieviele Tupel das Ergebnis der folgenden Algebraausdrücke jeweils enthält:

1. $R \bowtie S$
2. $S \bowtie R$
3. $R \setminus T$
4. $T \setminus R$
5. $R \cap T$
6. $T \cap R$
7. $R \div (\pi[B]S)$
8. $S \div (\pi[B]R)$

6 Aufgabe

(5 + 5 + 2 + 8 + 10 Punkte) Geben sie die beiden Tabellen *Conference* und *Country*, sowie die Ergebnis-Tabellen $E_1, E_2, E_3, E_4, E_5, E_6, E_7$ (null bezeichnet den SQL-Nullwert):

| Conference | | |
|------------|------|-----------|
| Name | Year | City |
| WWW | 2001 | null |
| ESWC | 2003 | Basel |
| ESWC | 2008 | Baroelona |
| SIGMOD | 2005 | Berlin |
| SIGMOD | 2006 | Paris |
| SIGMOD | 2007 | null |
| VLDB | 2005 | Berlin |
| VLDB | 2006 | Paris |
| VLDB | 2001 | Rome |
| ISWC | 2007 | Karlsruhe |
| ISWC | 2007 | Rome |
| ISWC | 2009 | Paris |
| PODS | 2001 | Basel |
| PODS | 2002 | Berlin |
| PODS | 200 | 4 Paris |
| PODS | 2005 | Rome |
| PODS | 2006 | Innsbruck |

| Country | |
|---------|---------|
| Name | Capital |
| Germany | Berlin |
| France | Paris |
| Italy | Rome |

| E_1 |
|-------|
| Name |
| VLDB |
| PODS |

| E_2 |
|-------|
| Name |
| ESWC |
| ISWC |
| PODS |

| E_3 |
|-------|
| Name |
| WWW |
| ESWC |

| E_4 | |
|------------|---------|
| Conference | Country |
| WWW | ESWC |

| E_5 |
|--------|
| Name |
| WWW |
| SIGMOD |
| VLDB |
| PODS |

| E_6 |
|--------|
| Name |
| WWW |
| ESWC |
| SIGMOD |
| ISWC |
| PODS |

| E_7 | |
|------------|---------|
| Conference | Country |
| SIGMOD | Germany |
| SIGMOD | France |
| VLDB | Germany |
| VLDB | France |
| VLDB | Italy |
| ISWC | Italy |
| ISWC | France |
| PODS | Germany |
| PODS | France |
| PODS | Italy |

| Antworten | |
|-----------|-----------------|
| Aufgabe | Ergebnistabelle |
| a) | |
| b) | |
| c) | |
| d) | |
| e) | |

Ordnen Sie den folgenden SQL-Anfragen das Ergebnis zu, das man erhält, wenn man die jeweiligen Anfragen auf *Conference* und *Country* anwendet. Schreiben Sie Ihre Lösungen in die obige Antworten-Tabelle.

1. SELECT DISTINCT name
FROM conference
WHERE city NOT IN (SELECT capital From Country);
2. SELECT DISTINCT name
FROM Conference a
WHERE NOT EXISTS
(
SELECT capital
FROM Country b
WHERE b .capital=a.city
);
3. SELECT name conference, name country
FROM Conference NATURAL JOIN Country;
4. SELECT name
FROM Conference
EXCEPT
SELECT name
FROM
(
SELECT name, capital
FROM
(SELECT name FROM Conferonce)
CROSS JOIN
(SELECT capital FROM Country)
EXCEPT
SELECT name, city
FROM Conference
);
5. SELECT DISTINCT name
FROM Conference a
WHERE NOT EXISTS
(
SELECT capital
FROM Country
WHERE capital NOT IN
(
SELECT city

```
FROM Conference b  
WHERE b.name=a.name  
)  
);
```

7 Aufgabe

(2 + 4 + 6 Punkte)

Geben sie jeweils eine BCNF-Zerlegung an und benenne Sie jeweils die Schlüssel des Ausgangsschemas und der Resultatsschemata:

1. $\mathcal{V} = \{A, B, C, D\}$
 $\mathcal{F} = \{A \rightarrow C, B \rightarrow D\}$
2. $\mathcal{V} = \{A, B, C, D, E\}$
 $\mathcal{F} = \{A \rightarrow C, AB \rightarrow E, B \rightarrow D\}$
3. $\mathcal{V} = \{A, B, C, D, E\}$
 $\mathcal{F} = \{A \rightarrow C, AB \rightarrow E, B \rightarrow D, D \rightarrow E\}$

Wenden Sie jeweils den BCNF-Analyse-Algorithmus an.

8 Aufgabe

(2 + 5 + 5 + 5 Punkte)

Auf einer Datenbanken sollen 3 Transaktionen zur Ausführung kommen.

1. Die Transaktionen haben die Form:

| | | |
|--------|----|----|
| $T_1:$ | RA | WA |
| $T_2:$ | RA | WA |
| $T_3:$ | RA | WA |

- a) Wieviele serielle Schedule gibt es zu T_1, T_2, T_3 ?
- b) Wieviele serialisierbare Schedule gibt es zu T_1, T_2, T_3 , die selbst nicht seriell sind?

2. Die Tansaktionen haben die Form:

| | | |
|--------|----|----|
| $T_1:$ | RA | WB |
| $T_2:$ | RB | WC |
| $T_3:$ | RC | WD |

- a) Wieviele nicht serielle Schedule gibt es zu T_1, T_2, T_3 ?
- b) ist gewährleistet, dass bei Anwendung eines Zeitmarken-Schedulers alle serialisierbaren Schedule zu T_1, T_2, T_3 zur Ausführung kommen können? Begründen Sie!

9 Aufgabe

(8 Punkte)

Zeigen oder widerlegen Sie die folgenden Aussage:

Jeder serialisierbare Schedule kann von einem 2PL-Scheduler bei geeigneter Wahl der Sperren und Freigaben ausgeführt werden.

Datenbanken und Informationssysteme

WS11/12

Prof. Dr. G. Lausen

16. März 2012

1 Aufgabe

(5 + 5 Punkte)

1. Betrachten Sie das Relationsschema **Verbindung(Von, Nach)**. Tupel in den Instanzen zu **Verbindung** repräsentieren Zugverbindungen zwischen Städten. Geben Sie eine SQL-Anfrage an, die alle Paare von unterschiedlichen Städten bestimmt zwischen den eine Verbindung mit einmal Umsteigen existiert.
2. Betrachten Sie das Relationsschema **Verbindung(Von, Nach, Dauer)**. Tupel in den Instanzen zu **Verbindung** repräsentieren jetzt Zugverbindungen zwischen Städten mit der betroffenen Fahrdauer. Geben Sie eine SQL-Anfrage an, die alle Paare von unterschiedlichen Städten bestimmt zwischen den eine direkte Verbindung existiert, die länger dauert, als alle Verbindungen mit einmal Umsteigen zwischen ihnen. Berücksichtigen Sie nur die Fahrtzeiten und nicht eventuelle Umsteigezeiten. Sie können annehmen, dass zu jeder direkten Verbindung auch mindestens eine Verbindung mit einmal Umsteigen existiert.

2 Aufgabe

(3+5+3+5 Punkte)

Seien die folgenden Ihnen aus der Vorlesung bekannten Relationsschemata gegeben:

Mitglied(LCode, Organisation, Art), Lage(LCode, Kontinent, Prozent)

Sie können davon ausgehen, dass die Instanzen von **Mitglieder** und **Lage** nicht leer sind.

1. Betrachten Sie die folgende SQL-Anfrage:

```
SELECT DISTINCT M1.LCode  
FROM Mitglieder M1, Mitglieder M2  
WHERE M2.LCode = 'A' AND M1.Organisation = M2.Organisation  
GROUP BY M1.Lcode  
HAVING COUNT(M1.Organisation) = ( SELECT COUNT(M3.Organisation)  
FROM Mitglied M3 WHERE M3.LCode = 'A' )
```

- a) Welches Ergebnis liefert die Anfrage? geben Sie eine Beschreibung in natürlicher Sprache.
- b) Geben Sie eine äquivalente Anfrage in der Relationenalgebra an. Zusätzlich zu den Basisoperatoren können Sie auch die aus der Vorlesung bekannten Operatoren \cap , \bowtie_α , \div verwenden.

2. Betrachten Sie die Folgenden SQL-Anfrage:

```
SELECT DISTINCT Organisation FROM Mitglied M  
WHERE NOT EXISTS (  
(SELECT LCode FROM Lage WHERE Lage.Kontinent - 'Europe') EXCEPT  
(SELECT LCode FROM Mitglied WHERE Organisation = M.Organisation))
```

- a) Welches Ergebnis liefert die Anfrage? geben Sie eine Beschreibung in natürlicher Sprache.
- b) Geben Sie eine äquivalente Anfrage in der Relationenalgebra an. Zusätzlich zu den Basisoperatoren können Sie auch die aus der Vorlesung bekannten Operatoren \cap , \bowtie_α , \div verwenden.

3 Aufgabe

(2+4 Punkte)

Sei gegeben die folgende Tabellendefinition:

```
CREATE TABLE R(  
A NIMERIC NOT NULL,  
B NIMERIC NOT NULL,  
C NUMERIC NOT NULL,  
PRIMARY KEY A);
```

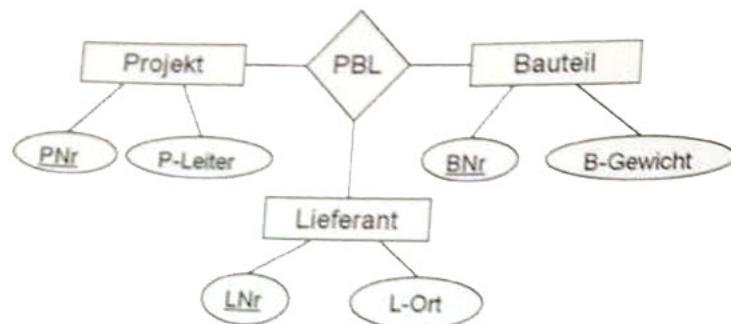
Die Instanzen von R seien frei von Duplikaten.

1. Geben Sie die CHECK-Klausel einer SQL-Tabellenbedingungen an, die für jede Instanz von R das Erfülltsein der folgenden Bedingung gewährleistet:
Wenn der Wert von B eines Tupels kleiner 0 ist, dann ist der Wert von C dieses Tupels größer als 0.
2. Geben Sie eine ASSERTION-Klausel in SQL an, die für jeden Instanz von R das Erfülltsein der funktionalen Abhängigkeit $B \rightarrow C$ gewährleistet.

4 Aufgabe

(3+3+6 Punkte)

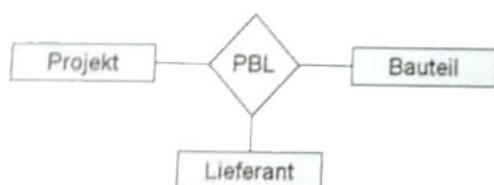
Betrachten Sie das folgende ER-Schema E:



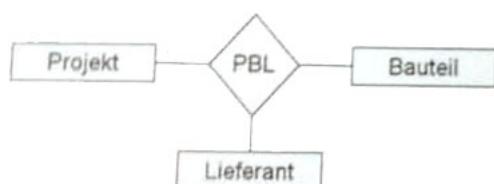
Das ER-Schema wird in unterschiedlichen relationalen Datenbank-Schemata transformiert.

Betrachten Sie die folgenden Varianten:

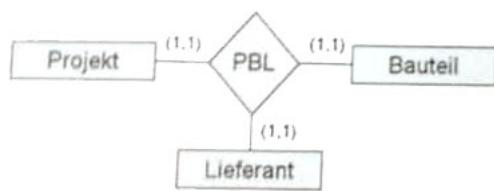
- Projekt(PNr, P-Leiter), Bauteil(BNr, B-Gewicht), Lieferant(LNr, L-Ort), PBL(PNr, BNr, LNr)
 - Projekt(PNr, P-Leiter), Lieferant(LNr, L-Ort), Bauteil(BNr, B-Gewicht, PNr, LNr)
 - PBL(LN, BNr, PNr, L-Ort, B-Gewicht, Pleiter)
1. Geben Sie Beziehungskomplexitäten für E an, so dass Variante a) äquivalent zu E ist und Variante b) nicht äquivalent zu E ist:



2. Geben Sie Beziehungskomplexitäten für E an, so dass Variante b) äquivalent zu E ist und Variante c) nicht äquivalent zu E ist:



3. Begründen Sie, dass für folgende Wahl der Beziehungskomplexitäten die Variante c) in 3NF ist. Geben Sie zunächst eine für Ihre Argumentation erforderliche Menge von funktionalen Abhängigkeiten an.



5 Aufgabe

(2+6 Punkte)

Betrachten Sie zu $V = \{A, B, C\}$ die folgende Menge von funktionalen Abhangigkeiten:

$$\mathcal{F} = \{A \rightarrow C, C \rightarrow B\}$$

1. Geben Sie zwei nichttriviale unterschiedliche funktionale Abhangigkeiten an, die nicht in \mathcal{F}^+ enthalten sind.
2. Geben Sie eine Instanz $r_{\mathcal{F}}$ so an, dass fur jede funktionale Abhangigkeit f uber V gilt:

$$r_{\mathcal{F}} \text{ erfullt } f \text{ genau dann, wenn } f \in \mathcal{F}^+.$$

$r_{\mathcal{F}}$ soll daruber hinaus nicht mehr als 4 Tupel enthalten.

6 Aufgabe

(4+2+2 Punkte)

Sei eine Relationsschema R gegeben mit

$$V = \{A, B, C, D\} \mathcal{F}\{A \rightarrow B, C \rightarrow D, BA \rightarrow A\},$$

1. Bestimmen Sie \mathcal{F}^{\min} . Sie können davon ausgehen, dass $A \rightarrow B$ und $C \rightarrow D$ in \mathcal{F}^{\min} enthalten sind - Sie brauchen also nur $BC \rightarrow A$ zu untersuchen.
2. Geben Sie mindestens zwei unterschiedliche Schlüssel an. Begründen Sie Ihre Lösung.
3. Geben Sie eine verlustfreie und abhängigkeitsbewahrende Zerlegung von R in 3NF an. wenden Sie hierzu den 3NF-Synthese-Algorithmus an. Begründen Sie die Ergebnisse der einzelnen Schritte des Algorithmus und kennzeichnen Sie die Schlüssel der Relationsschemata der Zerlegung.

7 Aufgabe

Seien Relationsschemata $R(A, B), S(B, C), T(C, D), Q(D, E)$ gegeben. Die Instanzen der Schemata enthalten keine Duplikate.

1. Der natürliche Verbund \bowtie ist kommutativ und assoziativ. Betrachten Sie dann $R \bowtie S$ und $(R \bowtie S) \bowtie R$. Sind die beiden Ausdrücke zueinander äquivalent? Beweisen Sie!
2. Es soll der natürliche Verbund über Relationen zu den Schemata R, S, T und Q berechnet werden. Geben Sie ein Beispiel für Left-Deep-Tree- und eine Bushy-Tree-Auswertung. Vermeiden Sie kartesische Produkte!
3. Es soll der natürliche Verbund über Relationen zu den Schemata zu den Schemata R, S und T berechnet werden. Wenn jede der beteiligten Relationen maximal 2 Tupel enthält, wieviel Tupel kann das Ergebnis des Verbundes maximal enthalten? Demonstrieren Sie Ihr Ergebnis anhand eines konkreten Beispiels.

8 Aufgabe

(2+3+3 Punkte)

Betrachten Sie das wie folgt beschriebene Hash-Join-Verfahren zur Berechnung des natürlichen Verbundes zwischen Relationen r und s . Sei N die Anzahl Seiten von r und sein M die Anzahl Seiten von s , wobei $N \leq M$. Die Relationen werden in Form von Blöcken gespeichert. Ein Block besteht aus einer Menge von Seiten. Die Anzahl der zur Verbundberechnung im Hauptspeicher zur Verfügung stehenden Seiten sei k .

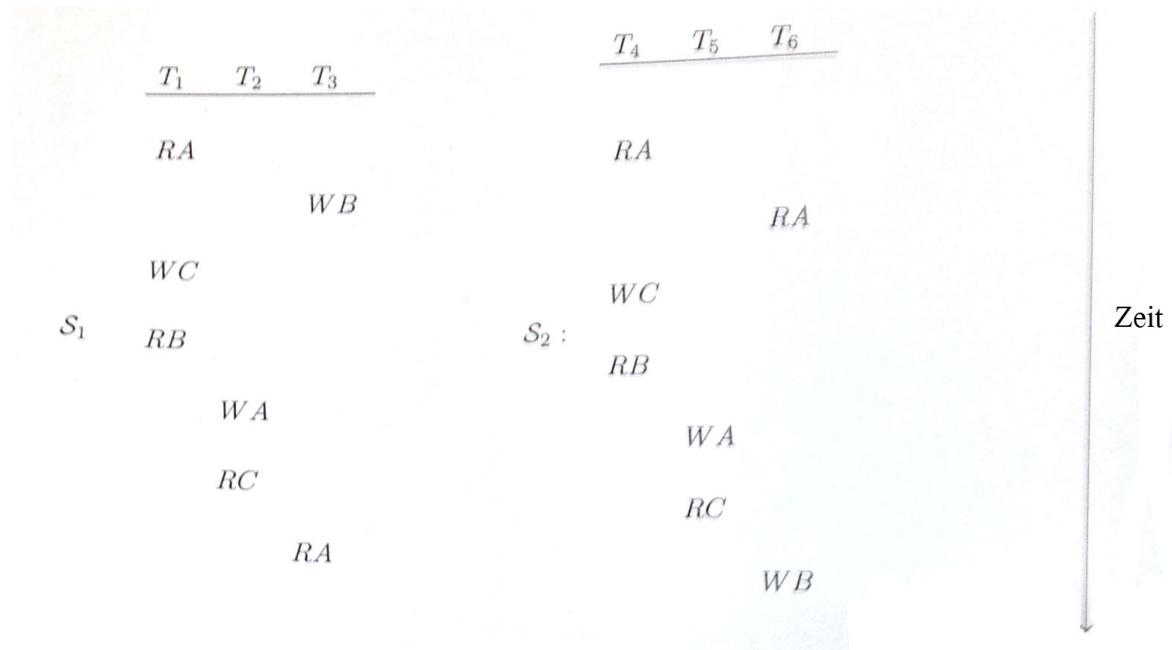
Sei h_1 eine Hashfunktion, die jedem Tupel einer Relation einen Block zuordnet. Sei h_2 eine von h_1 verschiedene Hashfunkiton, die jedem Tupel einer Relation einen Hauptspeicherbereich zuordnet.

1. Wende h_1 jeweils auf r und s an und bilde so je eine Partition von r und s .
2. Bilde dann den Verbund,
 - a) indem ein Block der Partition vor r gelesen wird um die Tupel in den Seiten des Blockes mittels h_2 im Hauptspeicher zu verteilen,
 - b) und dann die einzelnen Seiten des gemäß h_1 korrespondierenden Blockes der Partition von s nacheinander gelesen werden und für jedes Tupel einer Seite die korrespondierenden Tupel von r mittels h_2 bestimmt werden.
3. Iteriere das Verfahren bis alle Blöcke der Partition von r bearbeitet.
 - a) Warum sollten h_1 und h_2 unterschiedlich sein?
 - b) Welches Eigenschaften muss h_1 haben, damit Schritt (1.) nicht mehr als $2(N + M)$ Externzugriffe benötigt?
 - c) Welche Eigenschaften muss die durch Anwendung von h_1 erzeugte Partition von r haben, damit Schritt (2.) nicht mehr als $N + M$ Externzugriffe benötigt? Externzugriffe für das Schreiben des Endergebnisses des Verbundes sollen nicht berücksichtigt werden.

9 Aufgabe

(3+3+3+3 Punkte)

Betrachten Sie die folgenden Schedule jeweils dreier Transaktionen S_1 und S_2 :



1. Betrachten Sie S_1 .
 - a) Ist S_1 konflikt-serialisierbar? Begründen Sie!
 - b) Kann S_1 mittels 2-Phasen-Sperren realisiert werden? Begründen Sie! Falls ja, ergänzen Sie S_1 um die benötigten Sperr- und Freigabeoperationen LX und UX , wobei $X \in \{A, B, C\}$.
2. Betrachten Sie S_2 .
 - a) Ist S_2 konflikt-serialisierbar? Begründen Sie!
 - b) Kann S_2 mittels 2-Phasen-Sperren realisiert werden? Begründen Sie! Falls ja, ergänzen Sie S_2 um die benötigten Sperr- und Freigabeoperationen LX und UX , wobei $X \in \{A, B, C\}$.

**Klausur Datenbanken und Informationssysteme
WS 2012/13**

Name:

Vorname:

Matrikelnummer:

Studiengang: BSc. Info / MSc. Info / BSc. ESE / NF Info / Lehramt Info

Anmerkungen

- Tragen Sie Ihren **Namen** und Ihre **Matrikelnummer** auf diesem Deckblatt ein und markieren Sie Ihren **Studiengang**.
- Prüfen Sie die Klausur auf Vollständigkeit (7 Aufgaben).
- Zusätzliche Blätter, die Sie von uns erhalten können, müssen **Namen** und **Matrikelnummer** enthalten.
- Es sind keine Hilfsmittel zugelassen.
- Die Klausur dauert 90 Minuten.
- Mobiltelefone müssen ausgeschaltet sein.
- Zum Bestehen der Klausur sind mindestens 45 Punkte erforderlich.

| | maximale Anzahl Punkte | erreichte Anzahl Punkte |
|-----------|------------------------|-------------------------|
| Aufgabe 1 | 15 | |
| Aufgabe 2 | 15 | |
| Aufgabe 3 | 15 | |
| Aufgabe 4 | 12 | |
| Aufgabe 5 | 15 | |
| Aufgabe 6 | 8 | |
| Aufgabe 7 | 10 | |
| Σ | 90 | |

Aufgabe 1 (3+4+4+4=15 Punkte)

(1) Betrachten Sie die folgende Tabellendefinition zur Repräsentation von Freundschaftsbeziehungen:

```
CREATE TABLE Friendship (
    UserA_ID    NUMERIC,
    UserB_ID    NUMERIC);
```

- (a) Geben Sie eine SQL-Anfrage, die als Ergebnis die IDs aller User X liefert, die mit sich selbst in einer Freundschaftsbeziehung stehen, d.h., das Tupel (X, X) ist in der Tabelle enthalten
- (b) Geben Sie eine SQL-Anfrage an, die als Ergebnis alle Freundschaftsbeziehungen zweier beliebiger User X, Y liefert, die nicht symmetrisch sind, d.h., wenn (X, Y) in der Tabelle, dann ist (Y, X) nicht in der Tabelle.

(2) Betrachten Sie jetzt die folgende Tabellendefinition, in der zusätzlich der Zustand einer Freundschaftsbeziehung unterschieden wird. Eine aktive Freundschaft wird durch den Wert 1, eine passive durch den Wert 0 repräsentiert:

```
CREATE TABLE Friendship (
    UserA_ID    NUMERIC,
    UserB_ID    NUMERIC,
    State       NUMERIC);
```

Nehmen Sie zusätzlich an, dass alle Freundschaftsbeziehungen symmetrisch sind.

- (c) Geben Sie eine SQL-Anfrage an, die zu jedem User die Anzahl der aktiven Freundschaften bestimmt.
- (d) Geben Sie eine SQL-Anfrage an, die alle User liefert, die mehr aktive als passive Freundschaften besitzen.

Aufgabe 2 (3+3+3+6=15 Punkte)

(1) Seien $R(A, B), S(B, C), T(A, B)$ Relationsschemata. Betrachten Sie die folgenden Ausdrücke der Relationenalgebra:

- (a) $R \bowtie S \bowtie T \bowtie R$,
- (b) $T \bowtie R \bowtie S$,
- (c) $(R \cap T) \bowtie S$,
- (d) $(R \cup T) \bowtie S$.

Welche der folgenden Äquivalenzen sind korrekt? Geben Sie jeweils eine kurze Begründung.

$$(a) \equiv (b); (b) \equiv (c); (b) \equiv (d).$$

(2) Seien $R(A, B), S(B)$ zwei Relationsschemata. Widerlegen Sie anhand eines Gegenbeispiels die folgende Aussage:

$$(R \div S) \bowtie S \equiv R$$

Aufgabe 3 (5+5+5=15 Punkte)

Betrachten Sie die folgenden Tabellendefinitionen:

```
CREATE TABLE R (
A NUMBER,
B NUMBER);
```

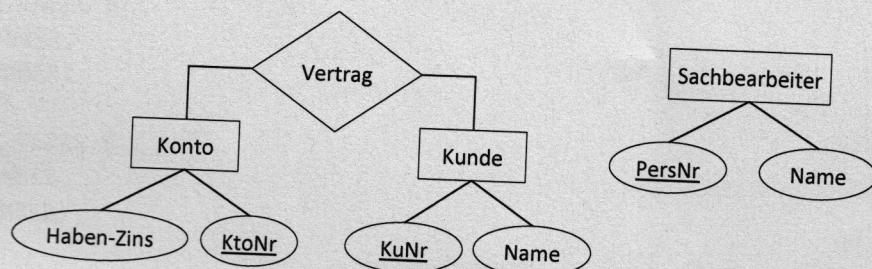
```
CREATE TABLE S (
C NUMBER,
A NUMBER);
```

Zu jedem A-Wert eines Tupels in S soll mindestens ein Tupel in R mit gleichem A-Wert existieren.

- (a) Geben Sie eine CHECK-Klausel für S an, die dies gewährleistet.
- (b) Geben Sie eine ASSERTION-Klausel an, die dies gewährleistet.
- (c) A sei der Primärschlüssel für Tabelle R. Geben Sie geeignete Klauseln für Primär- und Fremdschlüssel bzgl. Tabelle S an, die dies gewährleisten.

Aufgabe 4 (3+4+5=12 Punkte)

Betrachten Sie folgenden Ausschnitt eines ER-Diagramms, der Zusammenhänge in einer Bank modelliert:



Es sollen folgende Sachverhalte zusätzlich gelten:

- Kunden können mehrere Konten haben. Ein Kunde kann jedoch nur im System angelegt werden, wenn er mindestens ein Konto hat.
- Die Bank hat zwei Arten von Konten: Sparbücher und Girokonten, wobei Sparbücher einen zusätzlichen Höchstbetrag haben, der pro Monat abgehoben werden darf und Giro-Konten einen Soll-Zins haben.
- Ein Vertrag wird von genau einem Sachbearbeiter betreut.

Ergänzen Sie das ER-Diagramm so, dass diese Zusammenhänge ebenfalls repräsentiert werden. Geben Sie Beziehungskomplexitäten an und verwenden Sie die in der Vorlesung behandelten Erweiterungen des ER-Modells.

Aufgabe 5 (4+8+3=15 Punkte)

Gegeben sei ein Relationschema R über $V = \{A, B, C, D\}$ mit den funktionalen Abhängigkeiten

$$\mathcal{F} = \{A \rightarrow B, A \rightarrow C, C \rightarrow D, BC \rightarrow A, AD \rightarrow B\}.$$

- (1) Geben Sie alle Schlüssel an. Überprüfen Sie die Schlüsseleigenschaft (Eindeutigkeit und Minimalität) mittels des X^+ -Algorithmus.
- (2) Bestimmen Sie eine minimale Überdeckung \mathcal{F}^{min} der funktionalen Abhängigkeiten \mathcal{F} . Hinweis: untersuchen Sie hierzu zunächst $BC \rightarrow A$ und $AD \rightarrow B$ auf mögliche Linksreduktionen.
- (3) Ist R in 3NF? Begründen Sie. Falls R nicht in 3NF ist, geben Sie eine Zerlegung ρ von R mit Hilfe des 3NF-Synthese-Algorithmus an.

Aufgabe 6 (8 Punkte)

Gegeben seien die folgenden zwei Tabellen *Mitarbeiter* und *Abteilung*:

| | <i>PNr</i> | <i>Gehalt</i> | ... | <i>ANr</i> | |
|----------------------|------------|---------------|-----|------------|--|
| <i>Mitarbeiter</i> = | 4711 | 40000 | ... | 42 | |
| | ... | ... | ... | ... | |

| | <i>ANr</i> | <i>Name</i> | ... |
|--------------------|------------|-------------|-----|
| <i>Abteilung</i> = | 42 | ABC | ... |
| | ... | ... | ... |

Es soll folgende SQL-Anfrage ausgewertet werden:

```
SELECT *
FROM Mitarbeiter, Abteilung
WHERE Mitarbeiter.ANr = Abteilung.ANr
```

Es gelten folgende Annahmen:

- die Tabelle *Mitarbeiter* benötigt $M = 2000$ Seiten,
- die Tabelle *Abteilung* benötigt $A = 500$ Seiten,
- die Blockgröße b ist 100 Seiten groß,
- der Datenbankpuffer hat die Größe 102 Seiten.

Geben Sie die mindestens benötigte Anzahl der *lesenden* Seitenzugriffe bezogen auf obige Größen für die Berechnung der Anfrage unter Verwendung des *Block-Nested-Loop-Verbunds* an.

Aufgabe 7 (2+4+4=10 Punkte)

- (1) Betrachten Sie den Schedule

$$S = R_3 A R_1 B W_3 A R_1 A R_2 B W_1 B R_4 C W_4 C W_2 B.$$

Geben Sie den Konflikgraphen zu S an. Ist S serialisierbar? Begründen Sie!

- (2) Betrachten Sie die folgende Menge von Transaktionen \mathcal{T} :

$$\begin{aligned}T_1 &: RA WB \\T_2 &: RB WC \\T_3 &: RC WA\end{aligned}$$

Zeigen oder widerlegen Sie die folgende Aussage:

Jeder Schedule von \mathcal{T} ist serialisierbar.

- (3) Betrachten Sie die folgende Menge von Transaktionen \mathcal{T} :

$$\begin{aligned}T_1 &: RA WB WA \\T_2 &: RB WC WB \\T_3 &: RC WA WC\end{aligned}$$

Zeigen oder widerlegen Sie die folgende Aussage:

Wenn 2-Phasen Sperren für \mathcal{T} angewendet wird, dann kann kein Deadlock auftreten.

Klausur
Datenbanken und Informationssysteme
WS 2013/2014

Name:

Vorname:

Matrikelnummer:

Studiengang: BSc. Info / MSc. Info / BSc. ESE / NF Info / Lehramt Info

Anmerkungen

- Tragen Sie Ihren **Namen** und Ihre **Matrikelnummer** auf diesem Deckblatt ein und markieren Sie Ihren **Studiengang**.
- Prüfen Sie die Klausur auf Vollständigkeit (7 Aufgaben).
- Zusätzliche Blätter, die Sie von uns erhalten können, müssen **Namen** und **Matrikelnummer** enthalten.
- Es sind keine Hilfsmittel zugelassen.
- Die Klausur dauert 90 Minuten.
- Mobiltelefone müssen ausgeschaltet sein.
- Zum Bestehen der Klausur sind mindestens 45 Punkte erforderlich.

| | Punkte | erreichte Punkte |
|-----------|--------|------------------|
| Aufgabe 1 | 15 | |
| Aufgabe 2 | 15 | |
| Aufgabe 3 | 10 | |
| Aufgabe 4 | 10 | |
| Aufgabe 5 | 15 | |
| Aufgabe 6 | 10 | |
| Aufgabe 7 | 15 | |
| Σ | 90 | |

Aufgabe 2 (4+6+5=15 Punkte)

Betrachten Sie die folgende Tabellendefinition:

```
CREATE TABLE T (
    A    NUMBER NOT NULL,
    B    NUMBER NOT NULL );
```

und die folgende SQL-Anfrage Q :

```
SELECT DISTINCT A FROM T
EXCEPT
SELECT A FROM (
    SELECT A, B FROM (
        (SELECT A FROM T)
        CROSS JOIN
        (SELECT B FROM T WHERE A = 0)
    )
    EXCEPT
    SELECT A, B FROM T
);
```

- (a) Angenommen Q wird bzgl. der folgenden Instanz t von T ausgeführt. Welche Ausgabe liefert Q unter dieser Annahme?

| A | B |
|---|---|
| 0 | 0 |
| 1 | 0 |
| 0 | 1 |

- (b) Geben Sie zu Q einen äquivalenten Ausdruck der relationalen Algebra an. Sie können die folgenden Algebraoperatoren verwenden: σ, π, \div .
- (c) Betrachten Sie jetzt eine geänderte Definition der Tabelle T wie folgt:

```
CREATE TABLE T (
    A    NUMBER NOT NULL CHECK (A > 0),
    B    NUMBER NOT NULL CHECK (B > 0));
```

Geben Sie für diese geänderte Definition der Tabelle einen zu Q äquivalenten Ausdruck der relationalen Algebra an. Sie können die folgenden Algebraoperatoren verwenden: σ, π, \bowtie .

Aufgabe 3 (5+5=10 Punkte)

Betrachten Sie die folgende Tabellendefinition:

```
CREATE TABLE T (
    A    NUMBER,
    B    NUMBER );
```

und die folgende SQL-Anfrage Q :

```
SELECT COUNT(*) AS NUM
FROM (
    SELECT * FROM T
    WHERE A NOT IN
        (SELECT B FROM T)
);
```

- (a) Betrachten Sie die folgende Instanz t_1 von T:

| <i>A</i> | <i>B</i> |
|----------|-------------|
| 1 | 2 |
| 2 | <i>null</i> |
| 4 | 4 |
| 3 | 1 |

Geben Sie das Ergebnis von Q bzgl. t_1 an.

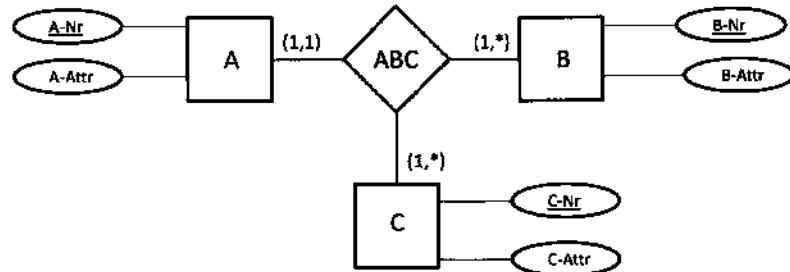
- (b) Betrachten Sie jetzt die folgende Instanz t_2 von T:

| <i>A</i> | <i>B</i> |
|-------------|----------|
| 1 | 2 |
| 2 | 2 |
| <i>null</i> | 4 |
| 3 | 1 |

Geben Sie das Ergebnis von Q bzgl. t_2 an.

Aufgabe 4 (5+5=10 Punkte)

Betrachten Sie das folgende ER-Diagramm:



- Geben Sie zu obigem ER-Diagramm ein äquivalentes ER-Diagramm mit maximal zwei Beziehungstypen an; diese Beziehungstypen müssen binär sein.
- Betrachten Sie den Beziehungstyp ABC als Relationsschema $R_{ABC}(A\text{-Nr}, B\text{-Nr}, C\text{-Nr})$. Nehmen Sie an, dass für dieses Schema außer den trivialen funktionalen Abhängigkeiten lediglich die funktionale Abhängigkeit $A\text{-Nr} \rightarrow B\text{-Nr}$ gilt. Begründen Sie, dass unter dieser Annahme die Beziehungskomplexität (1,1) in obigem ER-Diagramm falsch ist.

Aufgabe 5 (4+9+2=15 Punkte)

Gegeben sei ein Relationschema R über $V = \{A, B, C, D\}$ mit den folgenden funktionalen Abhängigkeiten:

$$\mathcal{F} = \{A \rightarrow B, A \rightarrow C, A \rightarrow D, C \rightarrow D, BC \rightarrow A, AD \rightarrow B\}.$$

- (a) Überprüfen Sie, ob A und BC Schlüssel sind, indem Sie mittels des X^+ -Algorithmus die Schlüsseleigenschaft (Eindeutigkeit und Minimalität) nachweisen.

- (b) Bestimmen Sie eine minimale Überdeckung \mathcal{F}^{min} zu \mathcal{F} .

Hinweis: Es genügt, wenn Sie hierzu $BC \rightarrow A$ und $AD \rightarrow B$ auf mögliche Linkss reduktionen sowie $A \rightarrow D$ auf mögliche Rechtsreduktionen überprüfen.

- (c) Ist R in 3NF? Begründen Sie!

Falls R nicht in 3NF ist, geben Sie eine Zerlegung ρ von R mit Hilfe des 3NF-Synthese-Algorithmus an.

Aufgabe 6 (4+6=10 Punkte)

Gegeben seien die folgenden zwei Tabellen *Employee* und *Department*:

| | <u>EmpID</u> | Salary | ... | DepID |
|-------------------|--------------|--------|-----|-------|
| <i>Employee</i> = | 4711 | 40000 | ... | 42 |
| | ... | ... | ... | ... |

| | <u>DepID</u> | Manager | Project | ... |
|---------------------|--------------|---------|---------|-----|
| <i>Department</i> = | 42 | Peter | P1175 | ... |
| | ... | ... | ... | ... |

Primärschlüssel sind durch Unterstrichen gekennzeichnet.

Es soll folgende SQL-Anfrage ausgewertet werden:

```
SELECT *
FROM Employee, Department
WHERE Employee.DepID = Department.DepID
```

Es gelten folgende Annahmen:

- die Tabelle *Employee* beinhaltet $t_E = 2000$ Tupel,
- die Tabelle *Department* beinhaltet $t_D = 200$ Tupel,
- ein Tupel von *Employee* sei 1 kB groß, ein Tupel von *Department* 2 kB,
- die Seitengröße s beträgt 10 kB.

- (a) Wie viele Tupel enthält das Ergebnis der Anfrage mindestens, wie viele maximal?
- (b) Angenommen die Tabellen sind jeweils bzgl. *DepID* sortiert. Geben Sie die mindestens und die höchstens benötigte Anzahl der *lesenden* Seitenzugriffe bezogen auf obige Größen für die Berechnung der Anfrage unter Verwendung des *Sort-Merge-Verbundes* an.

Aufgabe 7 (5+4+6=15 Punkte)

Betrachten Sie den Schedule

$$S = R_3A \ R_1B \ W_3A \ R_1A \ R_2B \ W_1B \ R_4C \ W_4C \ W_2C$$

- (a) Geben Sie den Konfliktgraphen für S an und beschriften Sie die Kanten des Graphen mit der Art des verursachenden Konflikts (RW, WR, WW). Ist S konflikt-serialisierbar?

Hinweis: Es genügt den nicht augmentierten Schedule zu betrachten.

- (b) Kann S bei Anwendung eines Zeitmarken-Schedulers, der die Zeitmarken relativ zur ersten Aktion einer Transaktion in einem Schedule vergibt, entstehen? Begründen Sie!
- (c) Betrachten Sie die folgende Menge von Transaktionen \mathcal{T} :

$$\begin{aligned} T_1 &: RA \ WB \\ T_2 &: RB \ WC \\ T_3 &: RC \ WA \end{aligned}$$

Zeigen oder widerlegen Sie, dass jeder Schedule von \mathcal{T} konflikt-serialisierbar ist.