

Klausur Datenbanken
Wintersemester 2010/2011
Prof. Dr. Wolfgang May
9. Februar 2011, 14-16 Uhr
Bearbeitungszeit: 90 Minuten

Vorname:

Nachname:

Matrikelnummer:

Studiengang:

Bei der Klausur sind **keine Hilfsmittel** (Skripten, Taschenrechner, etc.) erlaubt. Handies müssen ausgeschaltet sein. Papier wird gestellt. Benutzen Sie nur die **ausgeteilten**, zusammengehefteten **Blätter** für Ihre Antworten. Schreiben Sie mit blauem/schwarzem Kugelschreiber, Füller, etc.; Bleistift ist nicht erlaubt.

Zum **Bestehen** der Klausur sind **45** Punkte hinreichend.

- meine Note soll mit Matrikelnummer so bald wie möglich auf der Vorlesungs-Webseite veröffentlicht werden.
- meine Note soll nicht veröffentlicht werden; ich erfahre sie dann aus FlexNever oder beim zuständigen Prüfungsamt.

	Max. Punkte	Schätzung für "4"
Aufgabe 1 (ER-Modell)	20	15
Aufgabe 2 (Transformation in das Relationale Modell)	16	10
Aufgabe 3 (SQL und Relationale Algebra)	36	16
Aufgabe 4 (Verschiedenes)	18	8
Summe	90	49

Note:

Themenstellung: Wohnungsgenossenschaft

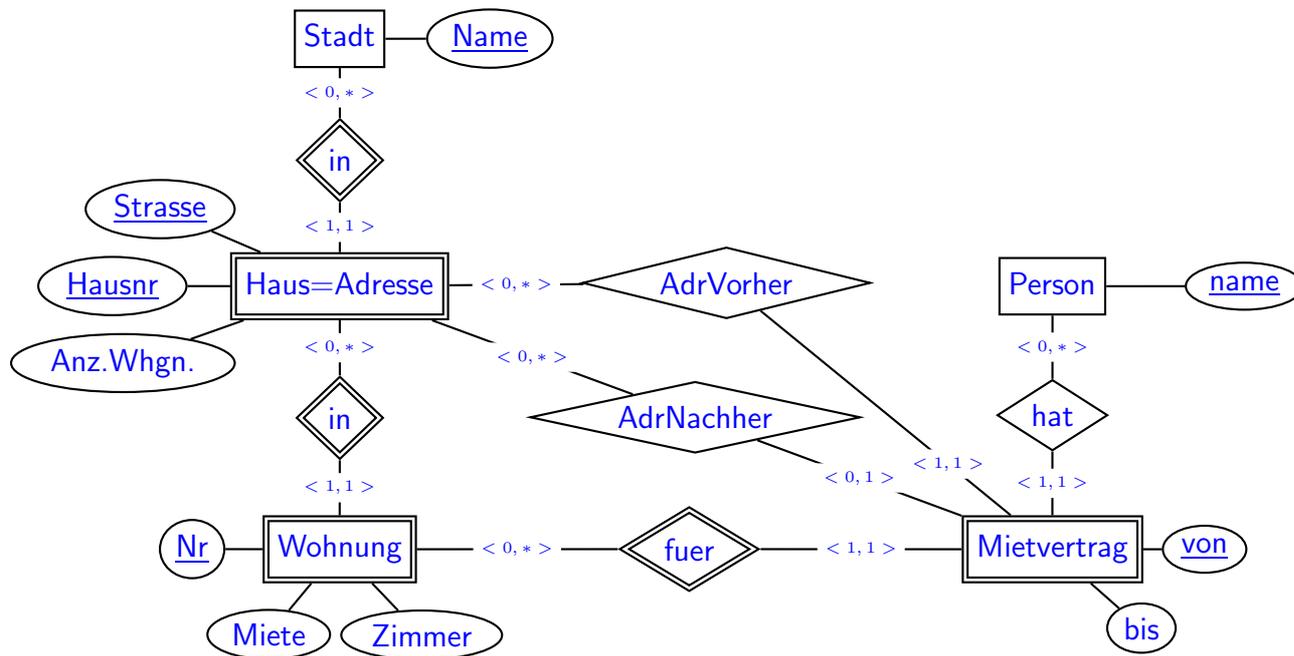
Alle Klausuraufgaben basieren auf einem gemeinsamen "Auftrag": In der Klausur soll eine Datenbank einer Wohnungsgenossenschaft, die Mietwohnungen in Mehrfamilienhäusern in verschiedenen Städten in ganz Deutschland besitzt, entworfen werden:

1. Es wird angenommen, dass Städte durch ihren Namen eindeutig identifiziert sind.
2. Zu den einzelnen im Besitz der Genossenschaft befindlichen Häusern sind Adresse (Strasse mit Hausnummer) und Stadt abgelegt. Ausserdem ist gespeichert, wieviele Wohnungen in dem Haus sind.
 - (a) Das Gebäude *Hauptstrasse 100* in *Göttingen* besteht aus 50 Wohnungen.
 - (b) Das Gebäude *Rheinstrasse 53* in *Köln* besteht aus 83 Wohnungen.
3. Jede Wohnung hat eine Nummer innerhalb des Hauses. Zu jeder Wohnung sind die Anzahl der Zimmer, die Quadratmeter und die monatliche (Kalt)Miete gespeichert.
 - (a) Die Wohnung mit der Nummer 42 des Hauses *Hauptstrasse 100* in *Göttingen* ist eine 3-Zimmer-Wohnung mit 80qm für 500 Euro im Monat.
 - (b) Die Wohnung mit der Nummer 43 desselben Hauses ist eine 2-Zimmer-Wohnung mit 50qm für 320 Euro im Monat.
 - (c) Die Wohnung mit der Nummer 17 des Hauses *Rheinstrasse 53* in *Köln* ist ein 1-Zimmer-Apartment mit 36qm und kostet 400 Euro im Monat.
4. Für jeden Mietvertrag ist gespeichert, wer welche Wohnung (und seit wann) gemietet hat (es wird angenommen, dass immer auf eine einzige Person als Mieter auftritt, d.h. den Mietvertrag unterschrieben hat). Ausserdem ist abgelegt, ob und zu welchem Zeitpunkt das Mietverhältnis gekündigt ist.
 - (a) *Karl Napf* wohnt seit 1.4.2008 in der Wohnung Nummer 42 des Hauses *Hauptstrasse 100* in *Göttingen*. Das Mietverhältnis ist nicht gekündigt.
 - (b) *Hans Dampf* wohnt seit 1.8.2001 in der Wohnung Nummer 43 des Hauses *Hauptstrasse 100* in *Göttingen*. Er hat die Wohnung zum 30.4.2011 gekündigt.
 - (c) *Lieschen Müller* wohnt seit 1.3.1980 in der Wohnung Wohnung Nr. 17 des Hauses *Rheinstrasse 53* in *Köln*. Sie hat die Wohnung zum 28.2.2011 gekündigt.
5. Wenn ein neuer Mietvertrag abgeschlossen wird, wird dabei auch die (zu dem Zeitpunkt noch aktuelle) Adresse des Mieters sowie der Beginn des Mietverhältnisses gespeichert.
 - (a) *Nils Pferd*, der zur Zeit in *Im Stall 4, Hintertupfing* wohnt (die Wohnung gehört nicht der Wohnungsgenossenschaft), hat einen Mietvertrag für die Wohnung Nummer 43 im Haus *Hauptstrasse 100* in *Göttingen* ab 1.5.2011 abgeschlossen.
 - (b) Der oben genannte *Hans Dampf* hat einen am 1.4.2011 beginnenden Mietvertrag für die Wohnung Nr. 17 im Haus *Rheinstrasse 53* in *Köln* abgeschlossen.
6. Wenn ein Mieter auszieht wird die neue Adresse gespeichert (z.B. für die Nebenkostenabrechnung des Auszugsjahres).
 - (a) Die neue Adresse von *Lieschen Müller* ist *Mühlengasse 24, Mühlhausen* in einem Haus, das nicht der Wohnungsgenossenschaft gehört.
7. Daten über alte Mietverträge werden nicht gelöscht.

Aufgabe 1 (ER-Modell [20 Punkte])

Entwickeln Sie ein ER-Modell für das Szenario. Geben Sie darin die Schlüsselattribute sowie die Beziehungskardinalitäten an.

Lösung



Hinweise:

- *Stadt* bzw. *Person* nur als Attribute zu *Haus* und *Mietvertrag* geht auch.
- *Von* ist Schlüsselattribut von *Mietvertrag*, um den Fall dass eine Person zweimal zu unterschiedlichen Zeiträumen in einer Wohnung wohnt, abbilden zu können (-1/2P). Damit muß die *hat*-Beziehung nicht notwendigerweise als identifizierend ausgezeichnet werden, d.h. *Wohnung* + *von*-Datum reicht als Schlüssel aus. (*Person* und *von*-Datum würden nur ausreichen, wenn man annimmt, dass eine Person nicht gleichzeitig zwei Mietverträge beginnt).
- *AdrVorher* und *AdrNachher* können auch als Attribute zu *Mietvertrag* gemacht werden (man verliert aber die in Aufgabe (3e) wertvolle Assoziation, dass dies ja ggf. auf ein Haus der Genossenschaft verweisen kann). Sinnvoll ist auch dann eine Aufteilung in *Stadt* und *Adresse*.
- Anstatt der expliziten Verwendung des Entitätstyps *Mietvertrag* könnte man einfach eine Beziehung *mietet(e)* zwischen *Person* und *Wohnung* mit den Attributen *von* und *bis* verwenden. Dieses würde das Konzept des Mietvertrages abdecken, es wäre aber dann nicht möglich, damit die vorher/nachher-Adressen zu verbinden (diese sind dann also nur Attribute zu *mietet(e)*).
- *Ganz schlecht* ist es, *AdrVorher* und *AdrNachher* mit *Person* zu assoziieren, da man so keinen (zeitlichen) Zusammenhang mit den einzelnen möglicherweise aufeinanderfolgenden Mietverträgen speichern kann.

Aufgabe 2 (Transformation in das Relationale Modell [16 Punkte])

- a) Lösen Sie diesen Aufgabenteil auf dem *letzten* Blatt und trennen dieses ab (und geben es am Ende mit ab!). Dann haben Sie dieses Blatt separat zugreifbar um später damit die Aufgaben 2b, und 3 (SQL, Relationale Algebra+SQLl) zu lösen.

Geben Sie an, welche Tabellen (mit Attributen, Schlüssel etc.) Ihre Datenbank enthält (keine SQL CREATE TABLE-Statements, sondern einfach grafisch). (9 P)

Markieren Sie dabei auch Schlüssel (durch unterstreichen) und Fremdschlüssel (durch überstreichen).

Geben Sie die Tabellen mit jeweils mindestens zwei Beispieldupeln (z.B. denen, die sich aus dem Aufgabentext ergeben, und weiteren erfundenen) an.

Lösung

- Tabellen für die Entitätstypen "Stadt" und "Person" werden nicht unbedingt benötigt, da sie nur eine Spalte "Name" enthalten würden.

Haus		
<u>Adresse</u>	<u>Stadt</u>	AnzWohnungen
Hauptstrasse 100	Göttingen	50
Rheinstrasse 53	Köln	83
:	:	:

(Strasse + Hausnummer in getrennten Spalten ist auch OK)

Wohnung					
<u>Adresse</u>	<u>Stadt</u>	<u>WhgNr</u>	Zimmer	Fläche	Preis
Hauptstrasse 100	Göttingen	42	3	80	500
Hauptstrasse 100	Göttingen	43	2	50	320
:	:	:	:	:	:
Rheinstrasse 53	Köln	17	1	36	400
:	:	:	:	:	:

Mietvertrag									
<u>Adresse</u>	<u>Stadt</u>	<u>WhgNr</u>	<u>von</u>	Person	<u>Adr.Alt</u>	<u>StadtAlt</u>	<u>bis</u>	<u>Adr.Neu</u>	<u>StadtNeu</u>
Hauptstr.100	Göttingen	42	1.4.2008	Karl Napf	null	null	null	null	null
Hauptstr.100	Göttingen	43	1.8.2001	Hans Dampf	null	null	30.4.2011	Rheinstr.53	Köln
Rheinstr.53	Köln	17	1.3.1980	L.Müller	null	null	28.2.2011	Mühleng.24	Mühlh.
Rheinstr.53	Köln	17	1.4.2011	Hans Dampf	Hauptstr.100	Göttingen	null	null	null
Hauptstr.100	Göttingen	43	1.5.2011	Nils Pferd	Im Stall 4	H'tupfing	null	null	null
:	:	:	:	:	:	:	:	:	:

- Man kann auch aufteilen:
PersonEinzug: Name, Adresse, Stadt, Datum, StadtAlt, AdresseAlt
PersonAuszug: Name, Adresse, Stadt, Datum, StadtNeu, AdresseNeu
 Wichtig ist, dass Name und Datum dabei sind, wenn jemand nacheinander mehrere Wohnungen mietet (oder auch dieselbe einige Zeit später nochmal).

- *StadtAlt*, *AdresseAlt* und *StadtNeu*, *AdresseNeu* sind keine Foreign Keys auf *Haus(StadtAdresse)*, da sie nur dann Referenzen sind, wenn die betreffenden Häuser auch im Besitz der Genossenschaft sind.
- b) Geben Sie das CREATE TABLE-Statement für diejenige Tabelle (bzw. die Tabellen), in der bei Ihnen die Daten über die Wohnungen abgespeichert sind, so vollständig wie möglich an (7 P).

Lösung

```

CREATE TABLE wohnung                                     Basis 4P
( adresse VARCHAR2(30),
  stadt  VARCHAR2(30),
  whgnr  NUMBER CHECK (whgnr > 0),
  zimmer NUMBER NOT NULL CHECK (zimmer > 0),           1/2 Checks
  flaeche NUMBER NOT NULL CHECK (flaeche > 0),         1/2 NOT NULL
  preis  NUMBER NOT NULL CHECK (preis > 0),
CONSTRAINT whgkey PRIMARY KEY (adresse, stadt, whgnr), 1P PKEY
CONSTRAINT whghaus FOREIGN KEY (adresse, stadt)
                REFERENCES haus(adresse,stadt) )      1P FKEY

```

Aufgabe 3 (SQL und Relationale Algebra [36 Punkte])

Verwenden Sie für diese Aufgabe die von Ihnen entworfene relationale Datenbasis. Keine der Antworten soll Duplikate enthalten.

- a) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck** an, die die Namen aller Städte ausgeben, in denen die Genossenschaft mindestens eine 4-Zimmer-Wohnung besitzt (2+2 P)

Lösung	<pre>select distinct stadt from wohnung where zimmer = 4;</pre>	$\begin{array}{c} \pi[\text{stadt}] \\ \\ \sigma[\text{zimmer}=4] \\ \\ \text{wohnung} \end{array}$
---------------	---	---

- b) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck** an, die die Namen aller Personen ausgeben, die gegenwärtig eine 4-Zimmer-Wohnung mit mindestens 100qm für weniger als 600E gemietet haben. Geben Sie auch einen soweit wie möglich (auf algebraischer Ebene) optimierten Algebra-Ausdruck an. (3+3+3 P)

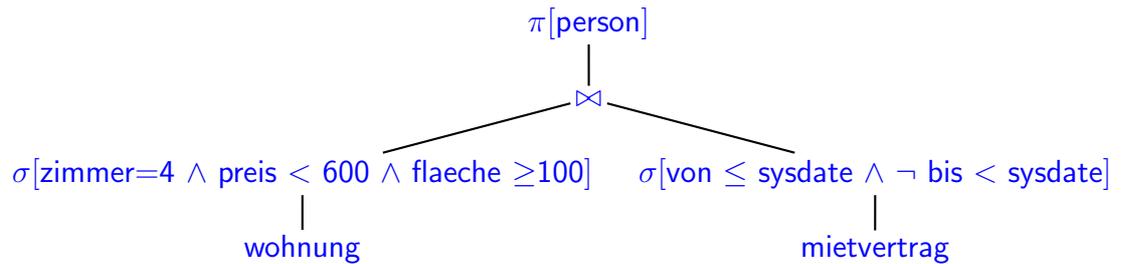
Lösung

```
select distinct person
-- distinct (-1/2P) ist wichtig, sonst sind alle Personen, die
-- gerade umziehen, mehrfach dabei
from mietvertrag mv, wohnung w
where mv.adresse = w.adresse
and mv.stadt = w.stadt
and mv.whgnr = w.whgnr
and w.zimmer = 4 and w.preis < 600 and w.flaeche >= 100
and mv.von <= sysdate and not bis < sysdate;
```

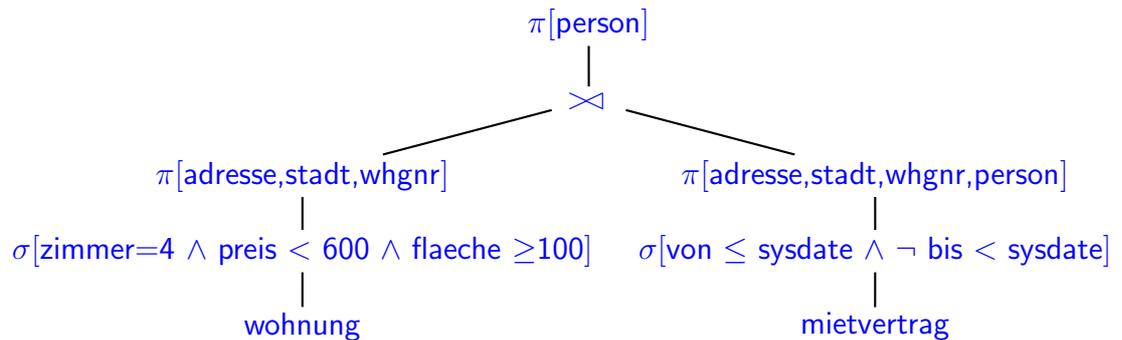
(beachten: bei gegenwaertigen Vertragen ist "bis" NULL oder > dem aktuellen Datum, und "von" < dem aktuellen Datum (vgl. Vertrag mit Nils Pferd, der schon gespeichert ist, aber erst am 1.5. beginnt) (-1/2 P wenn falsch)

```
select distinct person
from mietvertrag
where (adresse,stadt,whgnr) in
      (select adresse, stadt, whgnr
       from wohnung
       where zimmer = 4 and preis < 600 and flaeche >= 100)
and von <= sysdate and not bis < sysdate;
(oder "9.2.2011" direkt einsetzen)
-- geht auch ganz aehnlich mit ... where exists (... SFW ...)

-- der Algebra-Ausdruck entspricht der ersten Loesung:
```



Optimiert:



- c) Geben Sie **eine SQL-Anfrage** an, die für jede Stadt ausgibt, wieviel Miete die Genossenschaft insgesamt monatlich in dieser Stadt bekommen kann. (2 P)

Lösung

```
select stadt, sum(miete)
from wohnung
group by stadt;
```

- d) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck** an, der die Namen aller Personen ausgibt, die irgendwann in einer der Wohnungen der Wohnungsgenossenschaft gewohnt haben, und die aber nie in Köln in einer Wohnung der Wohnungsgenossenschaft gewohnt haben. (3+3 P)

Lösung

```
(select person
 from mietvertrag
 where von < sysdate)
minus
(select person
 from mietvertrag
 where Stadt = 'Koeln'
 and von < sysdate)
```

```
select person
from mietvertrag m1
where von < sysdate
and not exists
(select *
 from mietvertrag m2
 where von < sysdate
 and m1.person=m2.person)
```

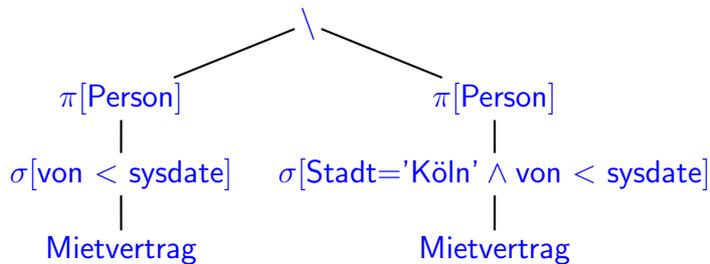
```

and m2.stadt = 'Koeln')

-- analog auch mit
-- where person not in (...)

-- das "von < sysdate" hatte fast niemand, gab ggf +1/2 Punkt.

```



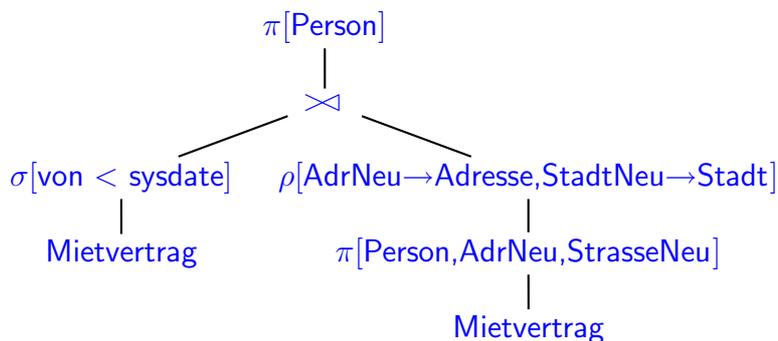
- e) Geben Sie **eine SQL-Anfrage und einen Algebra-Ausdruck** an, der die Namen aller Personen ausgibt, die mindestens einmal nach dem Auszug aus einer der Genossenschaft gehörenden Wohnung direkt in eine andere, auch der Genossenschaft gehörende Wohnung gezogen sind. (4 P)

Lösung

```

select distinct m.person
from mietvertrag m1, mietvertrag m2
where m.AdrNeu = m2.Adr
and m.StadtNeu = m2.Stadt
and m2.von < sysdate -- oder m1.bis < sysdate;
-- den Test auf sysdate (Umzug schon geschehen) hatte niemand;
-- haette auch hier wieder +1/2P gegeben;
-- haus anstatt mietvertrag.m2 war daher auch OK.

```



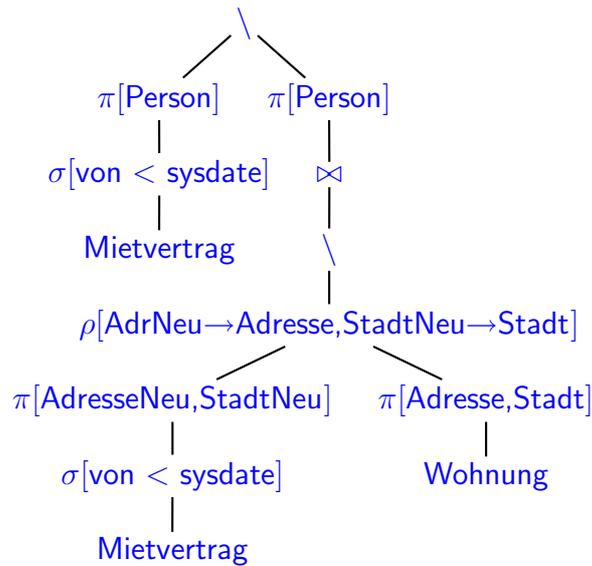
- f) Gibt es eine Möglichkeit, mit SQL aus der Datenbasis alle Personen herauszufinden, die seit ihrem erstmaligen Einzug ausschliesslich, ggf. aber beliebig oft, von einer der Genossenschaft gehörenden Wohnung direkt in eine andere, auch der Genossenschaft gehörende Wohnung gezogen sind?

Falls ja, geben Sie eine solche Anfrage (SQL oder Algebra) an. Falls nein, begründen Sie Ihre Antwort. (6 P)

Lösung Ja. Alle, die nie in eine nicht der Genossenschaft gehörende Wohnung umgezogen sind (also einschliesslich derer, die bisher noch nie ausgezogen sind); d.h. in etwa Negation von (e).

```
SELECT DISTINCT person
FROM mietvertrag m1 -- um alle bekannten Personen zu betrachten
WHERE m1.von < sysdate
WHERE NOT EXISTS -- Mietvertrag dieser Person, aus der diese Person
                  -- in eine nicht der G. gehörende Whg gezogen ist
  (SELECT *
   FROM mietvertrag m2
   WHERE m1.Person = m2.Person
        AND m2.von < sysdate
        AND bis IS NOT NULL
        AND -- Nachfolgewohnung gehört nicht der Genossenschaft
              (AdrNeu, StadtNeu) NOT IN
              (SELECT Adresse, Stadt
               FROM Wohnung))

(SELECT person
 FROM mietvertrag
 WHERE m1.von < sysdate)
MINUS
(SELECT person
 FROM mietvertrag
 WHERE von < sysdate
        AND bis IS NOT NULL
        AND -- Nachfolgewohnung gehört nicht der Genossenschaft
              (AdrNeu, StadtNeu) NOT IN
              (SELECT Adresse, Stadt
               FROM Wohnung))
```



- g) **Etwas Theorie:** Gegeben sind zwei Relationen $R(A, B, C, D)$ und $S(E, F, G, H)$. Die Attribute $R.C$ und $S.G$ sind numerisch, alle anderen sind Strings.

Geben Sie einen **Algebra-Ausdruck oder -Baum** an, der äquivalent zu der SQL-Anfrage

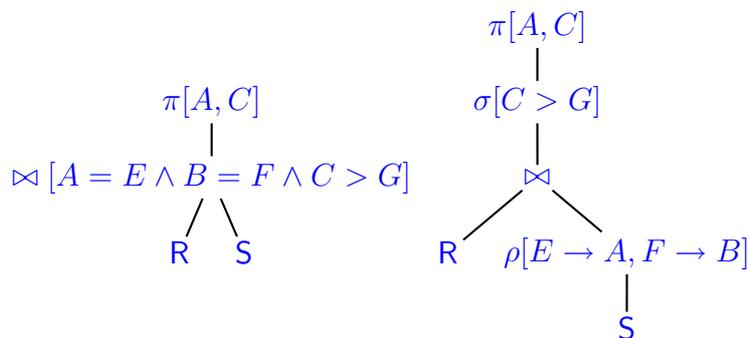
```

SELECT A, C
FROM R
WHERE (A, B) IN (SELECT E, F
                FROM S
                WHERE G < C)

```

ist. (5 P)

Lösung Alle Subqueries werden in Joins umgewandelt. Hier:



Aufgabe 4 (Verschiedenes [18 Punkte])

Diese Aufgabe bezieht sich auf die Kapitel “Anfrageauswertung und interne Strukturen”, “Transaktionen” und “Normalisierungstheorie” der Vorlesung. Die Antworten sollen kurz und klar formuliert sein; technische Details werden nicht erwartet.

- a) Gegeben sei eine Relation $R(A, B, C)$. Attribut B ist vom Typ `VARCHAR2(4)`. Es ist vorhersehbar, dass häufig Anfragen gestellt werden, in denen R in einem Teilausdruck der Form

```
SELECT ...
FROM R, ...
WHERE R.B = (SELECT ... FROM ... WHERE)
```

vorkommt.

Mit welchen Mitteln kann man diesen Zugriff unterstützen, so dass nicht jedesmal die gesamte Relation durchsucht werden muss? Geben Sie *kurz* an, wie dann der Zugriff erfolgt und welche Komplexität er hat. (4 P)

Lösung Zwei Möglichkeiten:

Man kann einen Baum (B^* -Baum) verwenden. Zugriff dann über Baumsuche in $O(\log n)$ (n ist die Anzahl der Tupel).

Alternative: Hash. Zugriff dann über Auswerten der Hashfunktion in $O(1)$.

- b) Transaktionen:

- 1) Was bedeutet die Eigenschaft *Durability (Dauerhaftigkeit)* von Transaktionen? (3 P)
- 2) Welchen Maßnahmen werden in kommerziell eingesetzten Datenbanksystemen eingesetzt, um dies zu gewährleisten? (3 P)

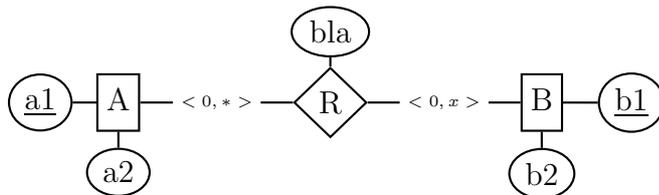
Lösung

- 1) Der Effekt einer *erfolgreich* beendeten Transaktion (z.B. die Bestätigung, dass eine Überweisung durchgeführt wurde) ist dauerhaft, d.h., (1) es wird nicht mehr durch ein Rollback oder ähnliches ungeschehen gemacht, und (2) auch im Fall eines Software- oder Hardware-Absturzes kann es nicht mehr verloren gehen.
- 2) Redundanz (wenn eine der Instanzen crasht, bleibt die andere bestehen), regelmäßige Backup-Kopien (Snapshots sichern), logfiles der ausgeführten Transaktionen und Aktionen (um auf einem Snapshot bzw. dem Absturzzustand aufbauend wieder einen konsistenten Zustand zu erreichen).

- c) Nehmen Sie an, dass Sie ein gutes ER-Diagramm entwickelt haben, und dabei sind, dies in ein relationales Modell umzusetzen.

- 1) In welchen Situationen benötigt man für einen Entitätstyp mehrere Relationen, um seine Informationen abzulegen? (2 P)

2) Gegeben sei folgendes ER-Diagramm



Geben Sie an, das Tabellenschema der Tabelle zur Speicherung von R im relationalen Modell an. (2 P)

Welche Attribute sind (u.a. in Abhängigkeit von x in der Kardinalität von B bzgl. R) Schlüsselattribute dieser Relation? (4 P)

Lösung

- 1) Wenn er ein oder mehrere mehrwertige Attribute hat.
(Antworten, die auf die Definitionen der 2. und 3. Normalform eingehen sind auch richtig, allerdings kann man diese Situationen durch ein wirklich gut durchdachtes ER-Modell in den allermeisten Fällen verhindern).
- 2) Die Relation R hat die Attribute a_1 , b_1 , bla (oder als A , B , bla benannt).
 $x = 1$: nur b_1 .
Wenn $x > 1$ und jedes B zu jedem A nur einmal in dieser Beziehung stehen kann, bilden a_1 und b_1 den Schlüssel.
Wenn $x > 1$ und ein B zu einem A auch mehrmals mit verschiedenen Werten von bla in Beziehung stehen kann, bilden a_1 , b_1 und bla den Schlüssel.
[Bewertung: eine Antwort gab 2P, jede weitere 1.5P; insgesamt konnten also 5P erreicht werden]

[Trennen Sie dieses Blatt am besten vor Beginn der Bearbeitung ab]
Lösen Sie hier Aufgabe 2a