UNIVERSITÄT DES SAARLANDES
Fachbereich 14 – Informatik
Prof. Dr.-Ing. G. Weikum
Dipl.-Inform. R. Schenkel

Datenbanksysteme WS 1999/2000

Musterlösung zur ersten Teilklausur

Aufgabe 1: Relationenalgebra und Tupelrelationenkalkül (1+1=2 Punkte)

Formulieren Sie die folgende Anfrage sowohl in der Relationenalgebra als auch im sicheren Tupelrelationenkalkül:

Wie lauten die Namen der Kunden, die alle Bestseller gekauft (oder zumindest bestellt) haben? Ein Bestseller sei dabei ein Buch, das mindestens 100.000mal verkauft worden ist.

Relationenalgebra:

BS:=π[isbn](σ[verkaufszahl≥100.000](Buecher))

Π[name](Kunden |x| (π[email,isbn](Verkäufe) ÷ BS))

Tupelrelationenkalkül:

{k.name | k∈ Kunden *and* ∀b:(b∈ Bücher *and* b.verkaufszahl≥100.000) ⇒
(∃v:v∈ Verkäufe *and* v.isbn=b.isbn *and* v.email=k.email) }

Aufgabe 2: SQL (2+2+2=6 Punkte)

Formulieren Sie die folgenden Anfragen in SQL:

a) Wie lauten die Namen der Kunden, die bei mindestens einem Bestseller (also einem Buch, das mindestens 100.000mal verkauft wurde) unter den ersten 100 Käufern waren?

```
SELECT name FROM kunden k, bücher b, verkäufe v
WHERE b.verkaufszahl>=100000 AND b.isbn=v.isbn
AND k.email=v.email AND v.kaufnr<=100;
```

b) Welcher Verlag unter den Verlagen, die mindestens zehn Bücher herausgebracht haben, hat die höchste Gesamtverkaufszahl (also Summe der Verkaufszahlen aller Bücher eines Verlages)?

```
CREATE VIEW verlagsverkäufe AS SELECT verlag,
SUM(verkaufszahl) AS verlagsverkaufszahl
FROM bücher GROUP BY verlag HAVING COUNT(*)>=10;

SELECT verlag FROM verlagsverkäufe
WHERE verlagsverkaufszahl>=ALL(SELECT verlagsverkaufszahl
                               FROM verlagsverkäufe);
```

c) Wie lauten die Namen der Kunden, die mehrere Exemplare eines Buchs auf einmal bestellt haben und die einzigen Käufer dieses Buches sind?

```
SELECT name FROM kunden k,verkäufe v WHERE v.anzahl>1
AND v.email=k.email AND NOT EXISTS
(SELECT * FROM verkäufe v1 WHERE v1.isbn=v.isbn
 AND v1.email<>k.email);
```

Aufgabe 3: Integritätsbedingungen in SQL (1+1=2 Punkte)

Formulieren Sie die folgenden Integritätsbedingungen als Assertion oder Trigger in SQL:

a) Ein Buch kann frühestens ab seinem Erscheinungsdatum geliefert werden.
Standardlösung (mit Assertion bzw. Check-Klausel):

```
CREATE TABLE verkaeufe(
/* ... */
  CREATE ASSERTION CHECK (Lieferdatum>=ALL(SELECT edatum
  FROM Buecher WHERE Buecher.isbn=isbn))
);
```

Alternativlösung mit Trigger:

```
CREATE OR REPLACE TRIGGER A3a BEFORE INSERT OR UPDATE
   OF lieferdatum,isbn ON verkaeufe REFERENCING NEW AS new
   FOR EACH ROW
DECLARE ed DATE;
BEGIN
  SELECT edatum INTO ed FROM Buecher b WHERE b.isbn=:new.isbn;
  IF :new.lieferdatum<ed
  THEN
    ROLLBACK WORK;
  END IF;
END;
```

b) Für einen Kunden mit einer offenen Bestellung (also einem Bestelldatum ungleich NULL, aber Lieferdatum gleich NULL) darf die Bankverbindung nicht geändert werden.

```
CREATE OR REPLACE TRIGGER A3b BEFORE UPDATE
   OF bankverbindung ON kunden
   REFERENCING NEW AS new FOR EACH ROW
DECLARE num_best LONG;
BEGIN
  SELECT COUNT(*) INTO num_best FROM Verkaeufe V WHERE
V.email=:new.email AND
  v.Bestelldatum IS NOT NULL AND v.Lieferdatum IS NULL;
  IF num_best>0
  THEN
    ROLLBACK WORK;
  END IF;
END;
```