

Gruppe A

Bitte tragen Sie **SOFORT** und **LESERLICH** Namen und Matrikelnr. ein und legen Sie Ihren Studentenausweis bereit.

PRÜFUNG AUS DATENBANKSYSTEME (181.038)			16. 3. 2000
Kennnr.	Matrikelnr.	Familienname	Vorname

Arbeitszeit: 120 Minuten. Die Aufgaben sind auf den Angabeblättern zu lösen; Zusatzblätter werden nicht gewertet.

Aufgabe 1:

(25)

Für einen Friedhof wird zur Arbeitserleichterung der Verwaltung eine Datenbank erstellt. Unterstreichen Sie je Relation einen Schlüssel. Verwenden Sie nur die vorgegebenen Attributnamen. (Diese sind nur bei ihrer jeweils ersten Erwähnung angeführt.) Führen Sie keine zusätzlichen Attribute ein und verwenden Sie möglichst wenige Relationen. Die Datenbank unterstützt keine Nullwerte. Für eine Relation, die nicht in 3NF ist, gibt es keine Punkte!

Auf dem Friedhof gibt es viele Gräber. Jedes Grab hat eine eindeutige Nummer (GNR), eine Lagebeschreibung (LAGE), einen Besitzer (BESITZER) und eine maximale Sarganzahl (MAXSARG). In einem Grab können sich mehrere Verstorbene befinden. Särge haben eine eindeutige Bestellnummer (BNR) und einen Hersteller (HERSTELLER).

Weiters gibt es Verstorbene, von denen die eindeutige Totenscheinnummer (TNR), das Sterbedatum (SDATUM), das Geburtsdatum (GDATUM), der Vorname (VORNAME) und der Nachname (NACHNAME) bekannt sind. Es ist auch bekannt, in welchem Sarg der Verstorbene liegt. Man kann die Position (POSITION) von Verstorbenen relativ zu einem anderen Verstorbenen im selben Grab angeben (z. B. kann ein Verstorbener rechts, links, unter ... einem anderen Verstorbenen liegen).

Es gibt 3 Friedhofsgärtner mit eindeutiger Sozialversicherungsnummer (SVNR), und Vor- und Nachnamen (VORNAME, NACHNAME), die für die Betreuung der Gräber zuständig sind. Es ist bekannt, welcher von den Gärtnern für ein bestimmtes Grab verantwortlich ist.

Man kann bei der Friedhofsgärtnerei verschiedene Dienstleistungen bestellen (z. B. Pflanzen setzen, Kerzen zu Allerheiligen, ...). Jede Dienstleistung wird durch eine Nummer (DNR), eine Beschreibung (BESCHREIBUNG) und einen Preis (PREIS) beschrieben. Bestellungen beziehen sich immer auf Gräber und Dienstleistungen, es wird das Datum (DATUM) angegeben und die Person (PERSON), die die anfallende Rechnung bezahlt. Für die Statistik wird mitprotokolliert, wann (DATUM) welcher Gärtner welche Dienstleistung bei einem Grab durchgeführt hat und wie viele Stunden (STUNDEN) er dafür gebraucht hat.

grab (<u>GNR</u> , LAGE, BESITZER, MAXSARG, SVNR)
verstorbene (<u>TNR</u> , VORNAME, NACHNAME, GDATUM, SDATUM, GNR, BNR)
sarg (<u>BNR</u> , HERSTELLER)
position (<u>TNR</u> , <u>TNR</u> , POSITION)
gaertner (<u>SVNR</u> , VORNAME, NACHNAME)
dienstleistung (<u>DNR</u> , BESCHREIBUNG, PREIS)
bestellung (<u>DNR</u> , <u>DATUM</u> , <u>GNR</u> , PERSON)
protokoll (<u>SVNR</u> , <u>DATUM</u> , <u>DNR</u> , <u>GNR</u> , STUNDEN)
()

Aufgabe 2:

(34)

In einer Relationalen Datenbank einer Weinkellerei werden folgende Informationen gespeichert:

```
wein(sorte, jg, temp, dauer)    preis(sorte, jg, jahr, preis)
regal(rnr, anzahl)             lager(fnr, rnr, sorte, jg)
```

Jeder Wein wird identifiziert durch seine Sorte *sorte* und den Jahrgang *jg*, beschrieben durch seine Lagertemperatur und Lagerdauer. Wenn der Wein einen Preis bekommen hat, wird das Jahr, in dem er gewonnen hat und der Name des Preises gespeichert. Bei den Regalen wird eine eindeutige Nummer *rnr* und die Anzahl der Flaschen gespeichert, die sie fassen. Weiters wird der Lagerort der Weinflaschen (*fnr*) gespeichert (*lager*).

a) Schreiben Sie eine Abfrage in SQL, die Regalnummer und Fassungsvermögen aller vollen Regale ausgibt, die mit einem einzigen Wein (gleiche Sorte und Jahrgang) gefüllt sind und wie lange der Wein gelagert wird. [8]

```
select r.rnr, r.anzahl, w.dauer
from regal r, wein w, lager l
where r.rnr = l.rnr
      w.sorte = l.sorte and
      w.jg = l.jg
group by r.rnr, r.anzahl, w.dauer, w.sorte, w.jg
having count(*) = r.anzahl;
```

b) Schreiben Sie eine Abfrage in SQL, die jene Weinsorten ausgibt, die in allen Jahren, in denen die Kellerei einen Preis gewonnen hat, ebenfalls einen Preis gewonnen haben. [10]

```
select p.sorte
from preis p
where not exists (select *
                  from preis p1
                  where p.sorte, jahr not in (select sorte, jahr
                  from preis));
```

c) Schreiben Sie die Abfrage in Beispiel 2b) in Relationaler Algebra.

[7]

$$\pi_{sorte,jahrpreis} \div \pi_{jahrpreis}$$

d) Schreiben Sie ein stratifiziertes Datalog-Programm, das im Prädikat `beste_sorte(SORTE)` jene Weinsorten ausgibt, für die gilt, dass sie in mehr als 5 aufeinanderfolgenden Jahren einen Preis gewonnen haben. Das Prädikat `SUM(X,Y,Z)` ($Z = X + Y$) steht zur Verfügung.

[9]

```

aufeinanderfolgend(SORTE, J1, J1) :- preis(SORTE, _, J1, _).
aufeinanderfolgend(SORTE, J1, J2) :- aufeinanderfolgend(SORTE, J1, J), preis(SORTE, _, J2, _),
SUM(J, 1, J2).

beste_sorte(SORTE) :- aufeinanderfolgend(SORTE, J1, J2), SUM(J1, 6, J2).
    
```

Aufgabe 3:

(7)

Gegeben sind die Relationenschemata: $p(ACD)$, $q(BCE)$, $r(BDE)$.

Optimieren Sie den folgenden algebraischen Ausdruck:

$$\sigma_{A < 4 \vee D = 3} \pi_{DAC} (\sigma_{B \neq 5} (p \bowtie q) \cup (r \bowtie \pi_{AC}(p)))$$

$$(\sigma_{A < 4 \vee D = 3}(p) \bowtie \pi_C \sigma_{B \neq 5}(q)) \cup \sigma_{A < 4 \vee D = 3}(\pi_D(r) \bowtie \pi_{AC}(p))$$

Aufgabe 4:

(15)

Bestimmen Sie für jedes der Relationenschemata $R_i(R_i, F_i)$, $i \in \{1, 2, 3\}$, alle Schlüssel. Geben Sie eine verlustfreie und abhängigkeits-treue Zerlegung in 3NF mit möglichst wenig Relationen an. Unterstreichen Sie in jeder Relation der Zerlegung einen Schlüssel.

Es gelte: $R_i = ABCDEF$, $i \in \{1, 2, 3\}$

$F_1 = \{AB \rightarrow EF, EF \rightarrow B, E \rightarrow C, D \rightarrow C\}$

$F_2 = \{AB \rightarrow C, B \rightarrow F, CEF \rightarrow D\}$

$F_3 = \{A \rightarrow CBD, EF \rightarrow B, CD \rightarrow EF\}$

RS	Schlüssel	Zerlegung in 3NF (Einen Schlüssel in jeder Relation unterstreichen)			
R1	ABD, ADEF	R1-1 <u>ABDEF</u>	R1-2 <u>EC</u>	R1-3 <u>DC</u>	R1-4
R2	ABE	R2-1 <u>CEFD</u>	R2-2 <u>BF</u>	R2-3 <u>ABC</u>	R2-4 <u>ABE</u>
R3	A	R3-1 <u>EFB</u>	R3-2 <u>CDEF</u>	R3-3 <u>ACD</u>	R3-4

Aufgabe 5:

(6)

Gegeben ist das Relationenschema ABCDEF und die Menge **F** von funktionalen Abhängigkeiten. Bestimmen Sie eine minimale Überdeckung.

$$F = \{B \rightarrow F, AF \rightarrow CD, C \rightarrow DE, AB \rightarrow EF, ED \rightarrow E\}.$$

$B \rightarrow F, AF \rightarrow C, C \rightarrow D, C \rightarrow E$

Aufgabe 6: Gegeben sei folgende Parallelausführung der Transaktionen $T1-T3$.

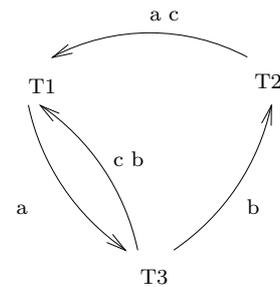
(7)

a) Geben Sie einen Präzedenzgraphen an:

[4]

$T1$	$T2$	$T3$
	read a	
	read c	
write a		write b
	read b	
read b		read c
		read a
write c		

Präzedenzgraph:



b) Ist die Ausführung serialisierbar ?

[3]

- ja, weil
- nein, weil der Präzedenzgraph zyklisch ist.

Die Ausführung wird aber nach Streichen von (mindestens) 1 Operation(en) serialisierbar, z.B. die Operation(en) read a in T3

Aufgabe 7:

(6)

Wieso werden die Sperrprotokolle pessimistische Verfahren, das Zeitstempelverfahren hingegen optimistisches Verfahren genannt?

[4]

Sperrprotokolle heißen pessimistische Protokolle, da sie den benötigten Datenbereich einfach sperren, auch wenn gar nicht klar ist, ob es zu einem Konflikt kommt oder nicht.

Das Zeitstempelverfahren ist ein optimistisches Protokoll, da alle Transaktionen starten und erst, wenn eine Konsistenzverletzung entdeckt wird, die Transaktion abgebrochen wird.

Wenn in einer Datenbank viele updates auf einem engen Datenbereich gemacht werden, welches Verfahren würden Sie zur Transaktionskontrolle verwenden und wieso?

[2]

Ein pessimistisches Verfahren, da es zu vielen Kollisionen durch Sperren der Daten eine bessere Performance erreicht werden kann als durch häufigen Transaktionsabbruch.

Gesamtpunkte: 100