

Gruppe A

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

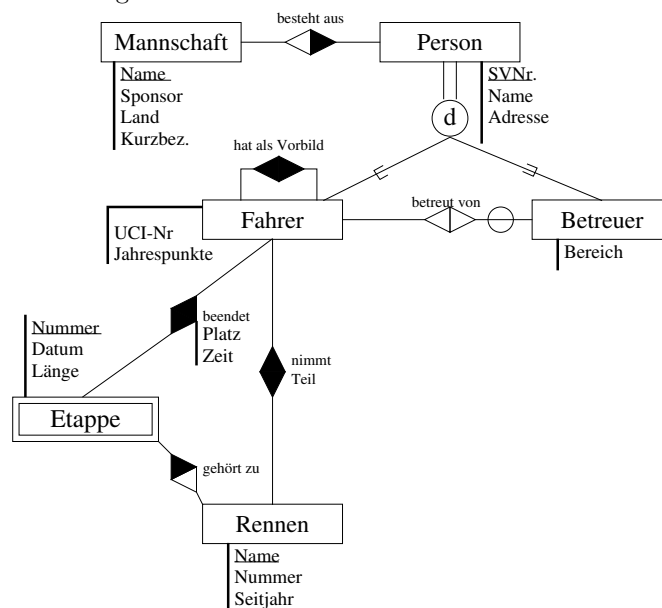
PRÜFUNG AUS DATENBANKSYSTEME (181.038)			9.10.2002
Kennnr.	Matrikelnr.	Familienname	Vorname

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

Aufgabe 1:

(20)

Erstellen Sie aufgrund folgenden EER-Diagramms ein Relationenschema in 3NF.



Unterstreichen Sie je Relation einen Schlüssel. Führen Sie keine zusätzlichen Attribute ein und verwenden Sie möglichst wenige Relationen. Die Datenbank unterstützt *Nullwerte*.

```

person ( SVNr, Name, Adresse, M.Name ) [2]
fahrer ( SVNr, UCI-Nr, Jahrespunkte, Betreuer.SVNr ) [2]
betreuer ( SVNr, Bereich ) [2]
mannschaft ( Name, Sponsor, Land, Kurzbez. ) [2]
rennen ( Name, Nummer, Seitjahr ) [2]
etappe ( Name, Nummer, Datum, Länge ) [3]
beendet ( SVNr, R.Name, Nummer, Platz, Zeit ) [3]
hatAlsVorbild ( SVNr, SVNr' ) [2]
nimmtTeil ( SVNr, Name ) [2]
( )

```

In der relationalen Datenbank eines Fußballverbandes werden folgende Informationen gespeichert:

spieler(nr, name, vnr) verein(vnr, name, liga)
spiel(spielnr, runde, jahr) ereignis(spielnr, minute, nr, typ)

Jeder Spieler wird durch eine Nummer (nr) eindeutig identifiziert und durch seinen Namen und seinen Verein beschrieben. Jeder Verein wird durch seine Vereinsnummer (vnr) eindeutig identifiziert und durch seinen Namen und die Liga, in der er spielt, beschrieben. Jedes Spiel wird durch eine Spielnummer (spielnr) eindeutig identifiziert und durch die gespielte Runde und das Jahr beschrieben. Während eines Spiels gibt es bestimmte Ereignisse, z.B. dass in der 23. Minute der Spieler mit der Nummer 31415 ein Tor geschossen hat (typ="tor").

Für jeden aktiv spielenden Spieler gibt es pro Spiel mindestens ein Ereignis, und wir gehen davon aus, dass es keine Eigentore gibt.

a) Torschützenliste: Schreiben Sie eine SQL-Abfrage, die für jeden Spieler seinen Namen, den Namen seines Vereins, sowie die Anzahl der von ihm im Jahr 2002 in der 12. Runde der Bundesliga erzielten Tore, sortiert in absteigender Reihenfolge, ausgibt. Wurden von einem Spieler weniger als 2 Tore erzielt, so soll er nicht in der Liste aufscheinen.

[10]

```
SELECT s.name, v.name, count(*)
FROM spieler s, verein v, spiel g, ereignis e
WHERE s.vnr=v.vnr AND s.nr=e.nr AND v.liga='Bundesliga' AND
      g.runde=12 AND g.jahr=2002 AND g.spielnr=e.spielnr AND
      e.typ='Tor'
GROUP BY s.name, v.name
HAVING count(*) > 1
ORDER BY count(*) DESC;
```

b) Schreiben Sie eine SQL-Abfrage, die die Namen der Vereine ausgibt, gegen die der „FC TUWien“ im Jahr 2002 noch nicht gespielt hat. Jeder Name soll höchstens einmal aufscheinen.

[10]

```
SELECT DISTINCT v.name
FROM spieler s, verein v, ereignis e
WHERE v.vnr=s.vnr AND s.nr=e.nr AND
      e.spielnr NOT IN (SELECT e1.spielnr
                        FROM spieler s1, verein v1, spiel g1, ereignis e1
                        WHERE v1.vnr=s1.vnr AND s1.nr=e1.nr AND
                              e1.spielnr = g1.spielnr AND
                              g1.jahr=2002 AND v1.name='FC TUWien');
```

c) Schreiben Sie eine SQL-Abfrage, die die Runden und Jahre ausgibt, in denen der „FC TUWien“ ein Spiel gewonnen hat.

[10]

```
SELECT DISTINCT g.runde, g.jahr
FROM spieler s, verein v, spiel g, ereignis e
WHERE v.name='FC TUWien' AND v.vnr=s.vnr AND s.nr=e.nr AND
      e.typ='Tor' AND g.spielnr=e.spielnr
GROUP BY e.spielnr, g.runde, g.jahr
HAVING count(*) >
      (SELECT count(*)
       FROM ereignis e1, spieler s1
       WHERE e.spielnr=e1.spielnr AND e1.nr=s1.nr AND
            s1.vnr<>s.vnr AND e1.typ='Tor');
```

Aufgabe 3:

(24)

Bestimmen Sie für jedes der Relationenschemata $R_i(R_i, F_i)$, $i \in \{1,2,3\}$, eine minimale Überdeckung.

Bestimmen Sie weiters alle Schlüssel und geben Sie eine verlustfreie und abhängigkeitstreue Zerlegung in 3NF mit möglichst wenig Relationen an. Unterstreichen Sie in jeder Relation der Zerlegung einen Schlüssel.

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

$F_1 = \{ AB \rightarrow BAD, D \rightarrow EF, F \rightarrow A, FAD \rightarrow C \}$

$F_2 = \{ AF \rightarrow C, CF \rightarrow A, C \rightarrow B, BC \rightarrow E, E \rightarrow D \}$

$F_3 = \{ A \rightarrow BCDE \}$

RS	minimale Überdeckung
R1	$AB \rightarrow D, D \rightarrow E, D \rightarrow F, F \rightarrow A, D \rightarrow C$
R2	$AF \rightarrow C, CF \rightarrow A, C \rightarrow B, C \rightarrow E, E \rightarrow D$
R3	$A \rightarrow B, A \rightarrow C, A \rightarrow D, A \rightarrow E$

RS	Schlüssel	Zerlegung in 3NF (Einen Schlüssel in jeder Relation unterstreichen)			
R1	AB, BD, BF	R1-1 <u>D</u> CE	R1-2 <u>A</u> BDF	R1-3	R1-4
R2	AF, CF ...	R2-1 <u>A</u> CF <u>A</u> <u>C</u> F	R2-2 <u>D</u> E	R2-3 <u>B</u> <u>C</u> E	R2-4
R3	AF	R3-1 <u>A</u> BCDE	R3-2 <u>A</u> F	R3-3	R3-4

Aufgabe 4:

(6)

Ist folgende Parallelausführung der Transaktionen $T1$ – $T4$ serialisierbar?

$T1$	$T2$	$T3$	$T4$
		read a	
		write a	
	read b		
read c			write d
		read b	
		write a	
			write d
write c		read a	
	read b		

☒ ja, weil b nur gelesen wird, und alle anderen Daten jeweils nur von einer einzigen Relation verwendet werden.

☐ nein, weil

Aufgabe 5:

(11)

Lässt sich jeder Ausdruck bestehend aus π und \bowtie (relationale Algebra) in eine gleichwertige SQL-Abfrage umwandeln [Ja/Nein]?

Begründen Sie Ihre Antwort (entweder durch ein Gegenbeispiel oder eine **grobe** Übersetzung)!

[6]

Ja. $\pi_{A_1, \dots, A_n} r$ bzw. $r \bowtie_{condition} s$ lassen sich durch

SELECT A_1, \dots, A_n FROM r ; bzw. SELECT * FROM r, s WHERE *condition*;
darstellen, und diese SELECT-statements lassen sich beliebig kombinieren.

Lässt sich jede Datalog-Abfrage eine gleichwertige Abfrage in relationaler Algebra umwandeln [Ja/Nein]?

Begründen Sie Ihre Antwort!

[5]

Nein; rekursive Abfragen sind nicht übersetzbar.

Aufgabe 6:

(8)

Gegeben sind drei Relationen $f(\underline{A}BC)$, $g(\underline{C}D)$, und $h(\underline{A}DCX)$ mit 2, 3, bzw. 4 Tupeln.

Geben Sie die minimale bzw. maximale Größe der durch die folgenden Ausdrücke entstehenden Relationen an:

Ausdruck	min. Ergebnisgröße	max. Ergebnisgröße
$f \bowtie f$	2	2
$f \bowtie g$	0	2
$f \bowtie (\pi_{CDX} h)$	0	8
$g \times g$	9	9

Gesamtpunkte: 99