

## Gruppe A

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

PRÜFUNG AUS DATENBANKSYSTEME (181.038)			25. 6. 2001
Kennnr.	Matrikelnr.	Familienname	Vorname

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

### Aufgabe 1:

(25)

Es ist die Datenbank eines Education Centers eines großen internationalen Konzerns zu modellieren. 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!

Das Education Center verfügt über eine Anzahl von Räumen. Jeder Raum hat eine eindeutige Raumnummer (RNR) und eine Telefonklappe (TEL).

Dazu gibt es zwei spezielle Kategorien von Räumen: Gastzimmer, bei denen zusätzlich noch die Anzahl von Betten (BETTEN) vermerkt ist, und Seminarräume, bei denen die Anzahl von Sitzplätzen (SITZE) sowie die Größe in  $m^2$  (M2) vermerkt sind.

Für jedes Seminar werden eine eindeutige Nummer (SEMNR), der Titel (TITEL), sowie das Beginn- und Endedatum (BEGINN, ENDE) gespeichert. Weiters sind für jedes Seminar ein oder mehrere Seminarräume sowie Gästezimmer gebucht, wobei Seminarräume nicht immer über den gesamten Zeitraum eines Seminars belegt werden, weshalb bei ihrer Buchung neben dem buchenden Mitarbeiter auch explizit Beginn und Ende (BBEGINN, BENDE) festgehalten werden.

Schließlich gibt es Mitarbeiter, für die jeweils Name (NAME), Geburtsjahr (GEBURT) und Geschlecht (SEX) gespeichert sind, sowie die eindeutige Mitarbeiternummer (MNR). Jedes Seminar wird von einem oder mehreren Mitarbeitern geleitet und von mehreren Mitarbeitern besucht. In letzterem Fall wird dabei auch noch die intern verrechnete Seminargebühr (GEBÜHR) angemerkt.

raum ( <u>RNR</u> , TEL	) [1+1]
gastzimmer ( <u>RNR</u> , BETTEN	) [1+1]
semraum ( <u>RNR</u> , SITZE, M2	) [1+1]
seminar ( <u>SEMNR</u> , TITEL, BEGINN, ENDE	) [2+1]
ggebucht ( <u>SEMNR</u> , <u>RNR</u>	) [2+1]
sgebucht ( <u>SEMNR</u> , <u>RNR</u> , <u>BBEGINN</u> , <u>BENDE</u> , MNR	) [3+1]
mitarbeiter ( <u>MNR</u> , NAME, GEBURT, SEX	) [2+1]
geleitet ( <u>SEMNR</u> , <u>MNR</u>	) [2+1]
besucht ( <u>SEMNR</u> , <u>MNR</u> , GEBÜHR	) [2+1]
(	)

## Aufgabe 2:

(34)

In der Relationalen Datenbank der TU Wien ist die folgende Information gespeichert:

```
studs(mnr, name, stknz, email)      absolviert(mnr, lvanr, datum, note)
lva(lvanr, titel, credits)          prereq(vorlvanr, lvanr)
```

Von Studierenden sind Matrikelnummer, Name, die inskribierte Studienkennzahl und die email Adresse bekannt. Von Lehrveranstaltungen werden die Nummer, der Titel und die Punkte gespeichert, die der Lehrveranstaltung zugeordnet sind. Weiters wird gespeichert, mit welchem Erfolg Studierende an Lehrveranstaltungen teilgenommen haben und welche Lehrveranstaltungen eine inhaltliche Voraussetzung (*vorlva*) für eine bestimmte andere Lehrveranstaltung darstellen.

a) Schreiben Sie eine Abfrage in SQL, die den Namen aller Studierender der Informatik (*stknz*=881) ausgibt, deren Notendurchschnitt unter 1,5 liegt und die im letzten Semester (WS00/01) Lehrveranstaltungen mit in Summe mehr als 30 Credits positiv absolviert haben. [10]

```
select s.mnr, s.name, avg(a.note)
from studs s, absolviert a
where s.mnr = a.mnr and
      s.stknz = 881 and
      s.mnr in (select a1.mnr
                 from absolviert a1, lva l1
                 where a1.lvanr = l1.lvanr and
                       a1.note < 5
                       a1.datum between '1.10.2000' and '31.1.2001'
                 group by a1.mnr
                 having sum(l1.credits) > 30)
group by s.mnr, s.name
having avg(a.note) < 1,5;
```

b) Schreiben Sie eine Abfrage in SQL, die Name und email Adresse aller Studierender ausgibt, die Lehrveranstaltungen absolviert haben, deren direkte Voraussetzung jedoch nicht, und die Titel all dieser Veranstaltungen. [9]

```
select s.mnr, s.name, s.email, l.titel
from studs s, absolviert a, lva l, prereq p,
where s.mnr = a.mnr and
      a.lvanr = l.lvanr and
      p.lvanr = a.lvanr and
      p.vorlvanr not in (select a1.lvanr
                          from absolviert a1
                          where s.mnr = a1.mnr);
```

c) Schreiben Sie für Abfrage b) eine Abfrage in Relationaler Algebra, wobei Sie nur Matrikelnummer und Lehrveranstaltungsnummer ausgeben müssen. [7]

$$\pi_{s.mnr,p.lvanr} \left[ \pi_{s.mnr,p.vorlva,p.lvanr} (bowtieabsolviert \bowtie prereq) - \pi_{s.mnr,p.vorlva,p.lvanr} ((bowtieabsolviert \bowtie prereq)[p.vorlva = a'.lva]absolviert') \right]$$

d) Schreiben Sie ein stratifiziertes Datalog-Programm, das im Prädikat **notDB(LVANR)** alle Lehrveranstaltungen angibt, die die Lehrveranstaltung Datenbanksysteme nicht als Voraussetzung haben (**Achtung:** auch indirekte Voraussetzungen sind zu betrachten) [8]

```
nachDB(LVANR) :- lva(LVADB, 'Datenbanksysteme', C),
                  lva(LVANR, T, C1), prereq(LVADB,LVANR).
nachDB(LVANR) :- nachDB(LVANR1), prereq(LVANR1,LVANR).

notDB(LVANR) :- not nachDB(LVANR), lva(LVANR, T, C).
```

### Aufgabe 3:

(7)

Gegeben sind die Relationenschemata:  $p(DE)$ ,  $q(ABCD)$ ,  $r(ABCD)$ . Optimieren Sie den folgenden algebraischen Ausdruck:

$$\pi_{BC} \sigma_{(B=b) \wedge (D < d)} \left[ \left( \pi_{ABCD} (q \bowtie p) \right) \cup (r \cap q) \right]$$

$$\left[ \pi_{BC} \left( \pi_{BCD} \sigma_{(B=b) \wedge (D < d)} (q) \bowtie \pi_D \sigma_{D < d} (p) \right) \right] \cup \left[ \pi_{BC} \left( \sigma_{(B=b) \wedge (D < d)} (r) \cap \sigma_{(B=b) \wedge (D < d)} (q) \right) \right]$$

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

$F_1 = \{AD \rightarrow E, B \rightarrow CD, E \rightarrow F\}$

$F_2 = \{F \rightarrow DE, E \rightarrow CB, B \rightarrow AF\}$

$F_3 = \{B \rightarrow CD, C \rightarrow A, A \rightarrow EFD, F \rightarrow A\}$

RS	Schlüssel	Zerlegung in 3NF (Einen Schlüssel in jeder Relation unterstreichen)					
R1	AB .....	R1-1 <u>EF</u> .....	R1-2 <u>ADE</u> .....	R1-3 <u>BCD</u> .....	R1-4 <u>AB</u> .....		
R2	F, E, B ..	R2-1 <u>F</u> ABCDE ...	R2-2 ..... ..	R2-3 ..... ..	R2-4 ..... ..		
R3	B .....	R3-1 <u>A</u> EFD .....	R3-2 <u>CA</u> .....	R3-3 <u>BC</u> .....	R3-4 ..... ..		

**Aufgabe 5:**

(6)

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

$$\mathbf{F} = \{A \rightarrow L, MXN \rightarrow OPL, L \rightarrow XY, NM \rightarrow A\}.$$

$$A \rightarrow L, MN \rightarrow O, MN \rightarrow P, L \rightarrow X, L \rightarrow Y, MN \rightarrow A$$

**Aufgabe 6:**

(7)

Die nachfolgende Parallelausführung von Transaktionen wird mit Hilfe von Zeitstempeln synchronisiert.

a) Geben Sie die Werte der Zeitstempel an (soweit möglich); markieren Sie einen Transaktionsabbruch mit '\*'. [4]

$T_1$	$T_2$	$T_3$	a	b	c
50	5	10	RT=WT=0	RT=WT=0	RT=WT=0
		read b		RT= 10	
	read b			RT= 10	
		read c			RT= 10
read a			RT= 50		
write a			WT= 50		
	write b			WT= *	
		write b		WT= 10	
write c					WT= 50
		write c			WT= 50

b) Welche Reihenfolge wird zwischen den Transaktionen durch die Zeitstempel festgelegt ( $T_i < T_j \equiv "T_i \text{ vor } T_j"$ )? Annahme: Eine abgebrochene Transaktion wird sofort nach dem Abbruch wieder gestartet. [3]

$$T_3 < T_1 < T_2$$

**Aufgabe 7:**

(6)

Geben Sie ein Beispiel, wann es sinnvoll (bzw. sogar notwendig) sein kann bei der Modellierung von 3.NF abzuweichen und begründen Sie Ihre Entscheidung.

Argumentieren Sie *nicht* mit Performance! Sie können folgende Relationen in Ihrem Beispiel verwenden (müssen dies aber nicht): `lieferung(kundennr, datum, adresse, telefonnummer)`, `kunde(kundennr, name, adresse, telefonnummer)`.

Erklärung anhand des Beispiels: Über die Zeit kann sich etwa die Kundenadresse ändern, beim neuerlichen Ausdruck von Lieferscheinen soll aber natürlich die tatsächlich belieferte Adresse aufscheinen.