

Gruppe A

Bitte **whlen** sie **welches Zeugnis Sie bentigen**, tragen Sie **sofort** und **leserlich** Namen und Matrikelnr. ein, und legen Sie Ihren Studentenausweis bereit.

PRÜFUNG AUS DATENBANKSYSTEME <input type="radio"/> 181.038 ("alt") <input type="radio"/> 181.146 ("neu") 15. 6. 2000			
Kennnr.	Matrikelnr.	Familienname	Vorname

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

Aufgabe 1:

(25)

Ein Weinliebhaber mit PC möchte seine Weine in einer Datenbank verwalten. 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!

Jeder Wein wird anhand der Bezeichnung (BEZEICHNUNG), des Winzers und des Jahrgangs (JAHRGANG) eindeutig identifiziert. Weiters wird gespeichert aus welchen Rebsorten der Wein gemacht wurde und wieviel Alkoholgehalt (ALK) er enthält, sowie wieviel Flaschen (ANZAHL) davon im Keller sind.

Von dem Winzer sind der Vorname (VORNAME), Nachname (NACHNAME), die Adresse (ADRESSE) und die Telefonnummer bekannt (TELEFON), wobei auszugehen ist, dass letztere sicher eindeutig ist.

Von den verschiedenen Rebsorten wird der eindeutige Sortenname (SORTENNAME), die bevorzugte Anbauregion, sowie welche Art von Wein (ART) daraus gemacht wird (Rot-, Weisswein, ...) gespeichert. Von der Anbauregion weiss man den kennzeichnenden Namen (NAME), die Anbaugrösse (GR), und die angebauten Rebsorten. Weiters soll bekannt sein, welche die nächstgelegenen anderen Weinbauregionen sind, sowie wie weit diese jeweils entfernt (KM) und in welcher Himmelsrichtung (HIMMELSRICHTUNG) wie Norden, Osten etc. sie liegen.

Es werden auch Verkostungen durchgeführt, bei denen die Weine bewertet werden. Für jeden Jahrgang werden dabei pro Rebsorte der beste und zweitbeste Wein aufgelistet, wobei davon auszugehen ist, dass immer ein erster und zweiter Platz vergeben werden. (Nur Weine, die aus einer Sorte bestehen, werden verglichen.)

wein (<u>bezeichnung,jahrgang,telefon</u> ,alk,anzahl)
winzer (<u>telefon</u> ,vorname,nachname,plz)
region (<u>name</u> ,gr)
sorte (<u>sortenname</u> ,art,name)
wein2sorte (<u>bezeichnung,jahrgang,telefon,sortenname</u>)
sorte2region (<u>sortenname,name</u>)
benachbart (<u>name1,name2</u> ,km,himmelsrichtung)
verkostung (<u>jahrgang,sortenname</u> ,bez1,telefon1,bez2,telefon2)
()
()

Aufgabe 2:

(34)

In einer Relationalen Datenbank eines großen Unternehmens werden folgende Informationen gespeichert:

```
mitarbeiter(svnr,name,geburt,anr,geschlecht,svnr_vor)    abteilung(anr,name,adresse)
           gehalt(svnr,von,bis,gehalt)
```

Jeder Mitarbeiter identifiziert sich anhand seiner Sozialversicherungsnummer und wird beschrieben durch seinen Namen, das Geburtsdatum, die Abteilung in der er arbeitet, das Geschlecht und den Verweis auf den direkten Vorgesetzten - wenn es keinen Vorgesetzten gibt, ist das ein Nullwert. Abteilungen haben eine eindeutige Abteilungsnummer und in der Gehaltstabelle wird die gesamte Gehaltsinformation (welches Gehalt von wann bis wann) der Mitarbeiter gespeichert. Beim aktuellen Gehaltseintrag ist der **bis**-Wert ein Nullwert.

a) Schreiben Sie eine Abfrage in SQL, die die Abteilungen mit dem höchsten aktuellen Durchschnittsgehalt selektiert und den Namen der Abteilung und das Durchschnittsgehalt ausgibt. [8]

```
select a.name, avg(g.gehalt)
from abteilung a, mitarbeiter m, gehalt g
where a.anr=m.anr and m.svnr=g.svnr and
      g.bis isnull
group by a.name, a.anr
having avg(g.gehalt) ≥ ALL select avg(g.gehalt)
                           from mitarbeiter m, gehalt g
                           where m.svnr=g.svnr and
                                 g.bis isnull
                           group by m.anr;
```

b) Schreiben Sie eine Abfrage in SQL, die von allen Mitarbeitern, die nach dem 1.1.1990 eingestellt wurden, den höchsten Gehaltssprung (zwischen zwei aufeinanderfolgenden Gehaltsstufen) berechnet und geben Sie Name, Sozialversicherungsnummer und den Gehaltssprung aus. [10]

```
select m.name, m.svnr, max(g1.gehalt-g2.gehalt)
from mitarbeiter m, gehalt g1, gehalt g2
where m.svnr=g1.svnr and m.svnr=g2.svnr and
      g1.von = g2.bis and
      m.svnr not in (select svnr from gehalt
                     where von < 1.1.1990)
group by m.name, m.svnr ;
```

c) Schreiben Sie die Abfrage in Relationaler Algebra, die den Namen und die Sozialversicherungsnummer jener Mitarbeiter ausgibt, die in der Mitarbeiterhierarchie in der ersten oder zweiten Stufe sind und den Namen der Abteilung, in der sie arbeiten. [7]

$$\pi_{a.name, m.name, svnr} \left[abteilung \bowtie \left(\pi_{anr, name, svnr}(mitarbeiter) - \pi_{anr'', name'', svnr''}(mitarbeiter_{[svnr_vor=svnr']} \bowtie mitarbeiter_{[svnr_vor'=svnr'']}) \right) \right]$$

d) Schreiben Sie ein stratifiziertes Datalog-Programm, das im Prädikat `nur_weibl_vor(SVNR, NAME)` jene Mitarbeiter ausgibt, die nur weibliche Vorgesetzte haben. [9]

```
vorgesetzt(SVNR1,SVNR2) :- mitarbeiter(SVNR1,N,G,A,S,SVNR2).
vorgesetzt(SVNR1,SVNR2) :- vorgesezt(SVNR1,SVNR), mitarbeiter(SVNR,N,G,A,S,SVNR2).

maennlich(SVNR)          := mitarbeiter(SVNR,N,G,A,'m',S).
hat_maennl_vor(SVNR)     :- vorgesezt(SVNR,SVNR1), maennlich(SVNR1).
nur_weibl_vor(SVNR,NAME) :- vorgesezt(SVNR,SVNR1), mitarbeiter(SVNR,NAME,G,A,S,SV),
                             non hat_maennl_vor(SVNR).
```

Aufgabe 3:

(7)

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

$$\sigma_{D=5 \wedge (A>6 \vee C<5)} \pi_{ACD} \left[\left(\pi_{ABC}(q) \cup p \right) \bowtie \left(r \cap \pi_{CDE}(q \bowtie r) \right) \right]$$

$$\left[\pi_{AC} \sigma_{A>6 \vee C<5}(q) \cup \pi_{AC} \sigma_{A>6 \vee C<5}(p) \right] \bowtie \left[\pi_{CD} \left(\sigma_{D=5} r \cap (\pi_{CD} \sigma_{D=5} q \bowtie \sigma_{D=5} r) \right) \right]$$

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 abhängigkeitsreue Zerlegung in 3NF mit möglichst wenig Relationen an. Unterstreichen Sie in jeder Relation der Zerlegung einen Schlüssel.

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

$F_1 = \{A \rightarrow C, E \rightarrow BD, B \rightarrow F, DB \rightarrow CE\}$

$F_2 = \{AD \rightarrow B, CB \rightarrow F, BF \rightarrow E\}$

$F_3 = \{CD \rightarrow AB, AB \rightarrow AE, AE \rightarrow CDF\}$

RS	Schlüssel	Zerlegung in 3NF (Einen Schlüssel in jeder Relation unterstreichen)			
R1	AE, ABD ..	R1-1 <u>AC</u>	R1-2 <u>BDC</u>	R1-3 <u>BF</u>	R1-4 <u>ABDE</u>
R2	ACD	R2-1 <u>BFE</u>	R2-2 <u>BCF</u>	R2-3 <u>ADB</u>	R2-4 <u>ACD</u>
R3	AB, CD, AE .	R3-1 <u>ABCDEF</u> ...	R3-2	R3-3	R3-4

Aufgabe 5:

(6)

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

$$\mathbf{F} = \{E \rightarrow AC, ACD \rightarrow F, AC \rightarrow D, D \rightarrow F, E \rightarrow F\}.$$

$$E \rightarrow A, E \rightarrow C, AC \rightarrow D, D \rightarrow F$$

Aufgabe 6:

(6)

Bei der Erweiterung des Wiederanlaufverfahrens nach einem Plattenfehler wird zugelassen, dass auch während des Backups Transaktionen aktiv sein können.

a) Ab welchem Zeitpunkt muss in diesem Fall das Logprotokoll vorhanden sein, um einen Wiederanlauf nach einem Plattenfehler durchführen zu können? [3]

Ab dem Zeitpunkt des Starts der ältesten während des Backups aktiven Transaktion.

b) Wird eine UNDO Liste benötigt und wenn ja, welche Transaktionen enthält sie? [3]

Ja, die UNDO Liste enthält jene Transaktionen, die zum Zeitpunkt des Backups und zum Zeitpunkt des Plattenfehlers aktiv waren.

Gesamtpunkte: 93