

1. Übungsblatt

3.0 VU Datenmodellierung

10. Oktober 2012

Allgemeines

In diesem Übungsteil werden Sie den Entwurf einer kleinen Datenbank üben, deren Überführung in das Relationenschema, sowie die relationale Algebra und den Relationenkalkül kennenlernen.

Lösen Sie die Beispiele eigenständig, denn bei der Prüfung und in der Praxis sind Sie auch auf sich alleine gestellt. Wir weisen Sie darauf hin, dass abgeschriebene Lösungen mit 0 Punkten beurteilt werden.

Geben Sie ein einziges PDF Dokument ab. Erstellen Sie Ihr Abgabedokument computerunterstützt. Wir akzeptieren keine gescannten handschriftlichen PDF-Dateien.

Deadlines

26.10. 06:55 Uhr Upload über den COURSEMANAGER
03.11. 12:00 Uhr Feedback im COURSEMANAGER verfügbar

Hinweise

1. Für dieses Übungsblatt werden maximal 5 Punkte vergeben.
2. Sie müssen mindestens einen Punkt auf die Abgabe bekommen. Wenn Sie weniger als einen Punkt auf das Blatt bekommen, oder kein Blatt abgegeben haben, sind Sie nicht zur ersten Leistungsüberprüfung zugelassen.
3. Im COURSEMANAGER ist ersichtlich, wieviele Punkte Sie auf die Abgabe bekommen haben. Weiters können Sie dort auf das Feedback zu Ihrer Abgabe zugreifen.

ER

Aufgabe 1 (ER Diagramm erstellen) [0.75 Punkte] Zeichnen Sie folgenden Sachverhalt in einem ER-Diagramm. Verwenden Sie dazu die (min, max)-Notation. Nullwerte sind nicht erlaubt. Eine Unterstützung bei der Erstellung von ER Diagrammen bietet das Tool dia (<http://live.gnome.org/Dia>). Achtung: im Diagramm Editor ER auswählen!

Der Arbeitsbereich für Databases und Artificial Intelligence plant ein System zur effizienten Verwaltung der LV *Datenmodellierung*. Es gibt Mitarbeiter und Studenten. Mitarbeiter werden eindeutig durch ihre Sozialversicherungsnummer (SVNR) identifiziert; Studenten eindeutig durch ihre Matrikelnummer (MNR). Zusätzlich wird bei beiden der Name (NAME) verwaltet. Bei Mitarbeitern ist außerdem noch eine Mailadresse (EMAIL) und das Geburtsdatum (GEBDAT) relevant. Zu Studenten wird ferner die Studienkennzahl (SKZ) und optional das Exmatrikulationsdatum (EXMATRIKULATIONS-DAT) angegeben. Von Mitarbeitern gibt es verschiedene Typen: Assistenten und Tutoren. Für Assistenten ist die Art der Anstellung (ANSTELLUNG) relevant; für Tutoren die Anzahl der Semester, die er/sie schon angestellt ist (ANZSEMESTER).

Es gibt zwei Übungsblätter pro Semester. Ein Übungsblatt wird eindeutig durch eine Nummer (NR) und das Semester (SEMESTER) identifiziert. Ferner ist bekannt, in welchem Zeitraum (VON, BIS) das Blatt zu bearbeiten ist. Die Abgabe eines Übungsblattes durch einen Studenten wird eindeutig durch das Übungsblatt, sowie den Studenten, dem die Abgabe zugeordnet ist, identifiziert. Zu jeder Abgabe werden der Zeitpunkt der Abgabe durch den Studenten (ZEITPUNKT) und die Abgabedatei (DATEI) gespeichert. Tutoren korrigieren die Abgaben und geben Feedback. Jede Abgabe wird von genau einem Tutor korrigiert. Dabei wird für die korrigierte Abgabe die Antwortdatei (FEEDBACK) abgespeichert, sowie die Anzahl der erreichten Punkte (PUNKTE).

Abgaben werden prinzipiell durch einen Plagiats-Checker und zusätzlich durch die Tutoren auf Plagiatsverdacht kontrolliert. Abgaben können also Plagiate beliebig vieler anderer Abgaben sein. Für ein vermutetes Plagiat wird ein Kommentar (KOMMENTAR) abgespeichert.

Assistenten prüfen Studenten nach der Abgabe der Übungsblätter in Form eines Prüfungsgespräches. Dabei prüft ein Assistent einen Studenten zu einem Übungsblatt. Im Zuge des Prüfungsgespräches müssen die erreichten Punkte (PUNKTE) und das Datum (DATUM) gespeichert werden.

In der LV gibt es zwei verschiedene Arten von schriftlichen Prüfungen: einen SQL-Test und eine Vorlesungsprüfung. Eine Prüfung wird eindeutig durch das Prüfungsdatum (DATUM) identifiziert. Studenten müssen bei einer Prüfung eine Mindestanzahl an Punkten erreichen (MINDESTPUNKTE). Ferner

gibt es zu jeder Prüfung eine maximale Punkteanzahl (MAXIMALPUNKTE). Für den SQL-Test ist zusätzlich noch relevant, ob dieser der Nachtragstest (NACHTRAGSTEST) ist. Für die Vorlesungsprüfung ist zusätzlich das Stoffgebiet (STOFFGEBIET) relevant. Um die Abhaltung von Prüfungen zu ermöglichen, müssen auch Hörsäle verwaltet werden. Ein Hörsaal wird eindeutig durch das Gebäude (GEBÄUDE) und eine Nummer (NUMMER) identifiziert. Zusätzlich sind der Standort (STANDORT) und die Anzahl der Sitzplätze (SITZPLÄTZE) bekannt. Mitarbeiter beaufsichtigen Prüfungen. Es wird gespeichert, welcher Mitarbeiter in welchem Hörsaal bei welcher Prüfung Aufsicht hat. Studenten treten zu Prüfungen an. Für jeden Antritt, sei es zum SQL-Test oder zur Vorlesungsprüfung, werden die erreichten Punkte (PUNKTE) gespeichert.

Am Ende des Semesters (bzw. nach den Prüfungen) werden Zeugnisse für Studenten ausgestellt. Ein Zeugnis ist dabei eindeutig durch den jeweiligen Studenten sowie das Ausstellungsdatum (AUSSTELLDATUM) identifiziert. Ein Student kann maximal drei Zeugnisse erhalten. Selbstverständlich muss zu jedem Zeugnis auch die Note (NOTE) gespeichert werden.

Lösung: Siehe Abbildung 1.

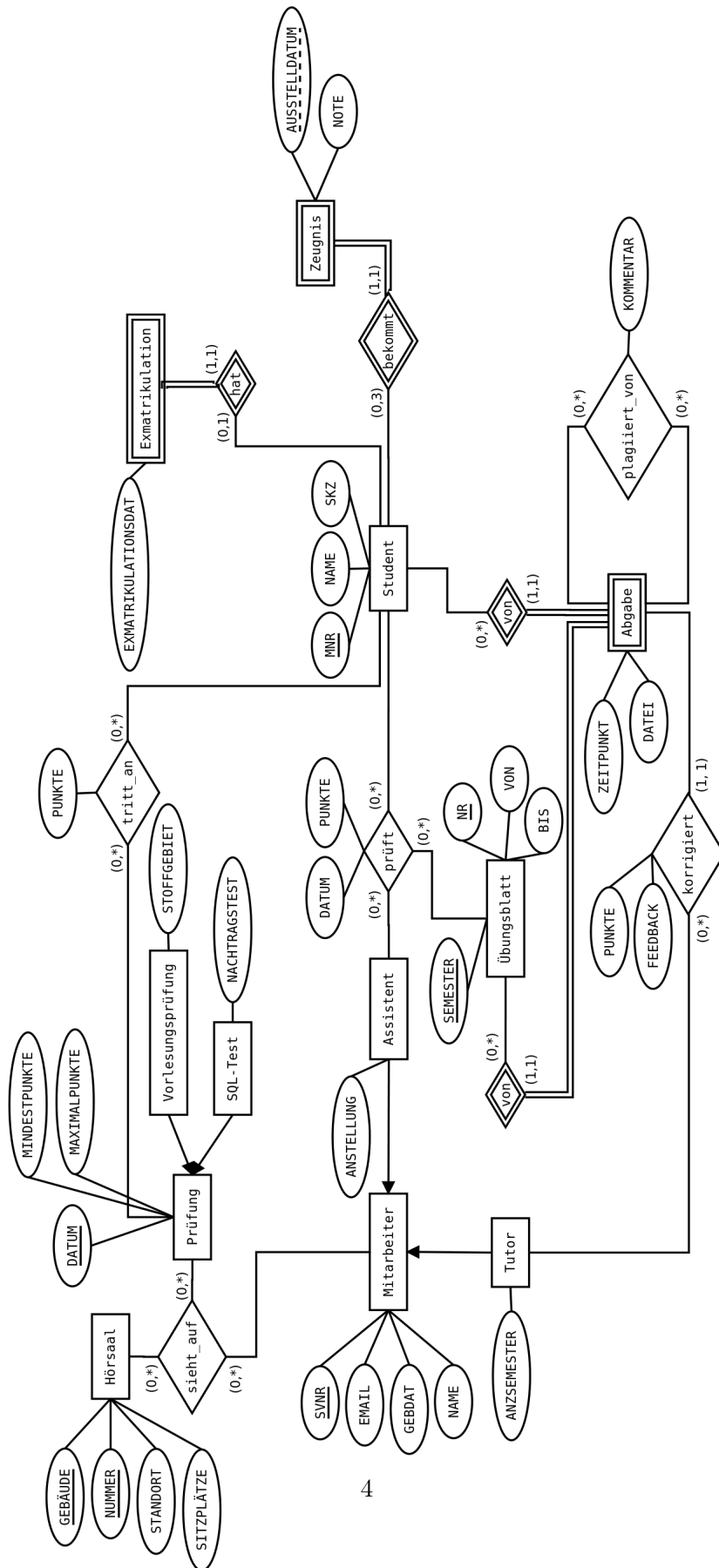


Abbildung 1: Lösung von Aufgabe 1

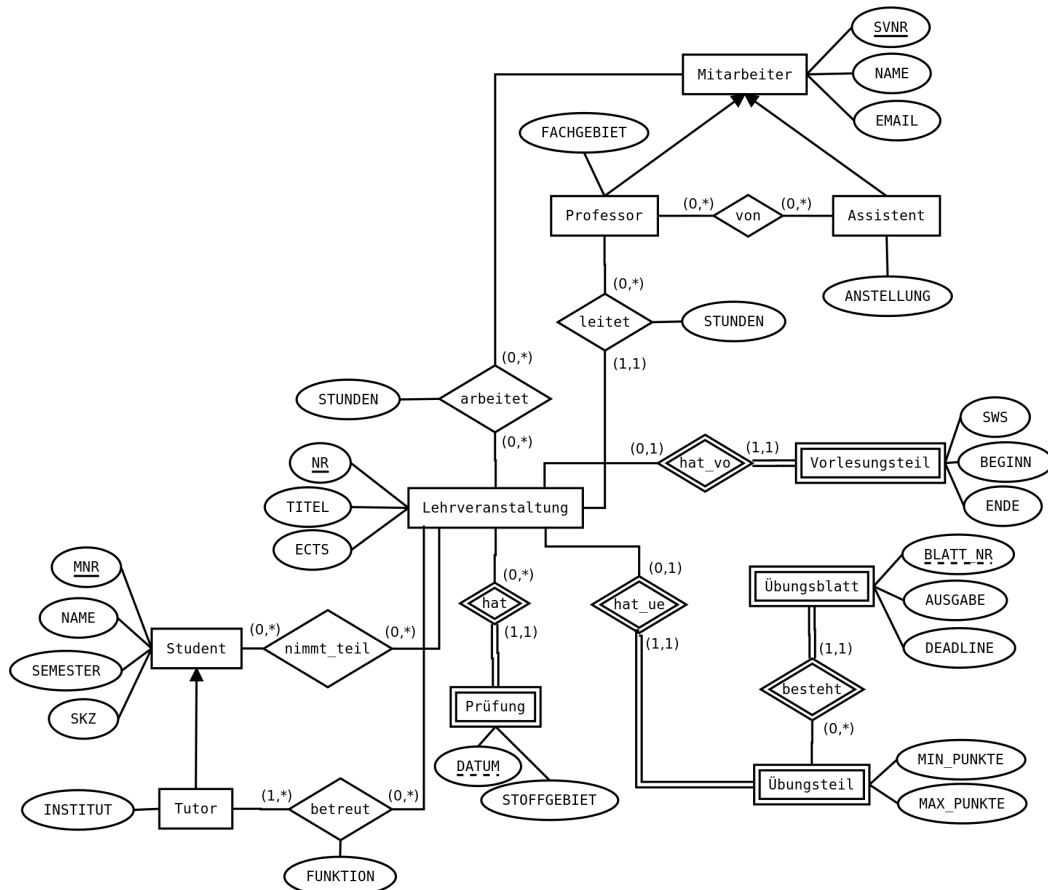


Abbildung 2: ER Diagramm zu Aufgabe 2

Aufgabe 2 (Überführung ins Relationenschema) [0.5 Punkte] Überführen Sie folgendes ER-Diagramm (Abbildung 2) ins Relationenschema. Nullwerte sind nicht erlaubt. Beachten Sie hierbei, dass Sie möglichst wenige Relationen verwenden. Unterstreichen Sie sämtliche Primärschlüssel, schreiben Sie die Fremdschlüssel kursiv und stellen Sie sicher, dass ein Fremdschlüssel eindeutig der passenden Relation zugeordnet werden kann.

Lösung:

- Mitarbeiter(SVNR, NAME, EMAIL)
- Professor(*Mitarbeiter.SVNR*, FACHGEBIET)
- Assistent(*Mitarbeiter.SVNR*, ANSTELLUNG)
- Lehrveranstaltung(NR, TITEL, ECTS, *Professor.SVNR*, STUNDEN)
- Student(MNR, NAME, SEMESTER, SKZ)
- Tutor(*Student.MNR*, INSTITUT)
- Vorlesungsteil(*Lehrveranstaltung.NR*, SWS, BEGINN, ENDE)
- Übungsteil(*Lehrveranstaltung.NR*, MIN_PUNKTE, MAX_PUNKTE)
- Übungsblatt(*Übungsteil.NR*, BLATT_NR, AUSGABE, DEADLINE)

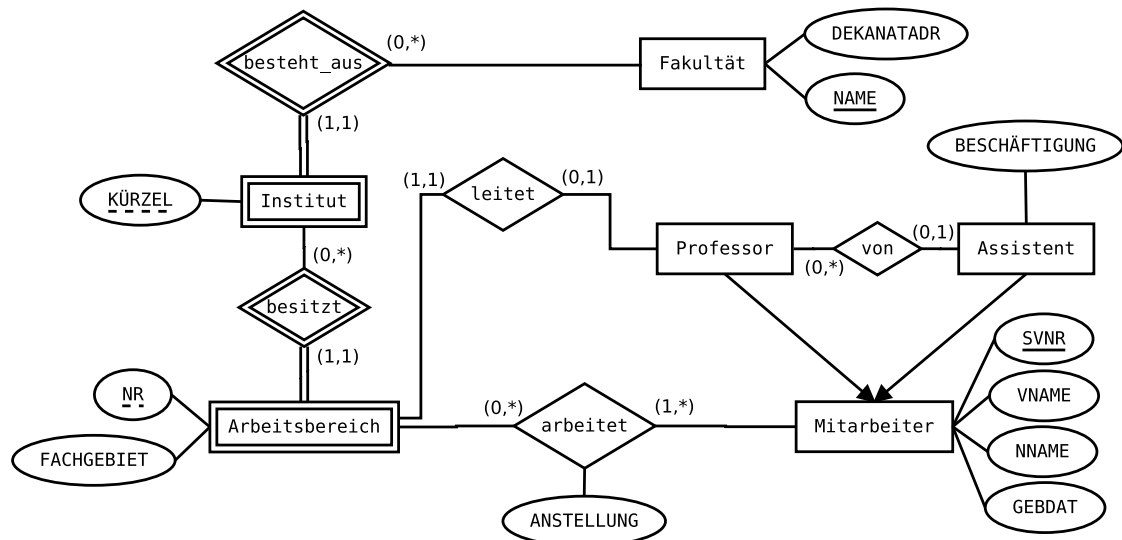


Abbildung 3: ER Diagramm zu Aufgabe 3

Prüfung(Lehrveranstaltung.NR, DATUM, STOFFGEBIET)
 nimmt_teil(Lehrveranstaltung.NR, Student.MNR)
 betreut(Lehrveranstaltung.NR, Tutor.MNR, FUNKTION)
 von(Professor.SVNR, Assistent.SVNR)
 arbeitet(Mitarbeiter.SVNR, Lehrveranstaltung.NR, STUNDEN)

Aufgabe 3 (Fehlersuche) [0.5 Punkte] Gegeben ist das folgende ER-Diagramm (Abbildung 3).

Dazu wurde das folgende (fehlerhafte !) Relationenschema erstellt:

Fakultät(NAME, DEKANATADR)
 Institut(Fakultät.NAME, KÜRZEL)
 Arbeitsbereich(Institut.KÜRZEL, NR, FACHGEBIET)
 Mitarbeiter(SVNR, VNAME, NNAME, GEBDAT)
 arbeitet(Arbeitsbereich.NR, Institut.KÜRZEL, Fakultät.NAME, ANSTELLUNG)
 Professor(SVNR, Arbeitsbereich.NR, Institut.KÜRZEL, Fakultät.NAME)
 Assistent(Mitarbeiter.SVNR, BESCHÄFTIGUNG, Professor.SVNR)

Finden Sie mindestens 4 Fehler, die sich bei der Überführung des ER-Diagramms ins Relationenmodell eingeschlichen haben, und korrigieren Sie das Relationenschema entsprechend. Wie lautet das korrekte Relationenschema? Beachten Sie, dass NULL-Werte nicht erlaubt sind.

Lösung:

Folgende Fehler wurden korrigiert:

- Beim Entitätstyp **Fakultät** sind **NAME** sowie **DEKANATADR** eindeutig. Ein Schlüssel muss minimal sein, also darf nur eines der beiden Attribute als Schlüssel fungieren. Wir entscheiden uns hier für **NAME**.
- Der schwache Entitätstyp **Arbeitsbereich** wurde nicht korrekt abgebildet, da zur eindeutigen Identifizierung auch der Schlüssel von **Fakultät** notwendig ist. Weiters wurde **Institut.KÜRZEL** nicht als Schlüsselattribut gekennzeichnet, sodass ein **Arbeitsbereich** einzig und allein durch die Nummer identifiziert werden würde.
- In der Beziehung **arbeitet** wurde **ANSTELLUNG** als Schlüsselattribut aufgenommen. Dies ist jedoch nicht notwendig. Ferner wurden die Schlüsselattribute für **Mitarbeiter** vergessen.
- Beim Entitätstyp **Professor** wurden die Attribute von **leitet** aufgenommen. Jedoch leitet nicht jeder Professor eine Fakultät und somit würden NULL-Werte entstehen. Es muss das Schlüsselattribut von **Professor** in **Arbeitsbereich** aufgenommen werden. Ferner verweist die Sozialversicherungsnummer bei **Professor** nicht auf **Mitarbeiter**.
- Beim Entitätstyp **Assistent** wurden die Attribute der Beziehung **von** aufgenommen. Ein Assistent muss jedoch nicht Assistent von einem Professor sein. Somit würden wiederum NULL-Werte entstehen. Die Beziehung **von** muss separat als Relation eingeführt werden. Ferner wurde hier **Professor.SVNR** als Schlüsselattribut aufgenommen, was jedoch der Minimalität des Schlüssels widerspricht.

Korrigiertes Relationenmodell:

```

Fakultät(NAME, DEKANATADR)
Institut(Fakultät.NAME, KÜRZEL)
Arbeitsbereich(Institut.KÜRZEL, Fakultät.NAME, NR, Professor.SVNR,
FACHGEBIET)
Mitarbeiter(SVNR, VNAME, NNAME, GEBDAT)
arbeitet(Arbeitsbereich.NR, Institut.KÜRZEL, Fakultät.NAME, Mitarbeiter.SVNR
ANSTELLUNG)
Professor(Mitarbeiter.SVNR)
Assistent(Mitarbeiter.SVNR, BESCHÄFTIGUNG)
von(Professor.SVNR, Assistent.SVNR)

```

Relationale Algebra - Relationenkalkül

Um Ihnen die Erstellung Ihrer Abgabe zu den folgenden Aufgaben zu erleichtern, haben wir unter <http://dbai.tuwien.ac.at/education/dm/resources/symbols.html> eine Liste mit den wichtigsten Symbolen der relationalen Algebra zusammengestellt. Sie können diese per copy/paste in Ihr Word/OpenOffice/... Dokument einfügen. Zusätzlich sind die entsprechenden L^AT_EX Befehle vermerkt.

Aufgabe 4 (Primitive Operatoren) [0.5 Punkte] Drücken Sie die Operationen Join (\bowtie), Durchschnitt (\cap) und Semi-Join (\ltimes) mit Hilfe der primitiven Operatoren (σ , Π , \times , ρ , $-$, \cup) aus.

Lösung:

Es wird angenommen, dass zwei Relationen S und T gegeben sind. Hierbei umfasst das Schema von S $g + s$ viele Attribute ($S_1, \dots, S_s, G_1, \dots, G_g$) und das Schema von T $g + t$ viele Attribute ($T_1, \dots, T_t, G_1, \dots, G_g$).

(a) $S \bowtie T \equiv \Pi_{S.G_1, \dots, S.G_g, S_1, \dots, S_s, T_1, \dots, T_t} (\sigma_{S.G_1=T.G_1 \wedge \dots \wedge S.G_g=T.G_g} (S \times T))$

(b) $S \cap T \equiv S - (S - T)$

(c) $S \ltimes T \equiv \Pi_{S.G_1, \dots, S.G_g, S_1, \dots, S_s} (\sigma_{S.G_1=T.G_1 \wedge \dots \wedge S.G_g=T.G_g} (S \times T))$

Aufgabe 5 (Äquivalenzen) [0.5 Punkte] Gegeben sind folgende relationale Ausdrücke über $R(\underline{A}B)$, $S(\underline{B}AC)$ und $T(\underline{D}BC)$. Überprüfen Sie, ob die folgenden Ausdrücke in relationaler Algebra äquivalent sind. Falls ja, so erklären Sie warum. Falls nein, geben Sie eine Begründung an.

(a) $r_1 : \Pi_A(R) \bowtie \Pi_C(T)$ und $r_2 : \Pi_{AC}(R \bowtie T)$

(b) $r_3 : S \ltimes \Pi_{BC}(T)$ und $r_4 : T \ltimes S$

(c) $r_5 : \sigma_{B=13 \vee B=31} (R \cup \Pi_{AB}(S))$ und $r_6 : \sigma_{B=13}(R) \cup \sigma_{B=31}(\Pi_{AB}(S))$

(d) $r_7 : \Pi_A(R) \times \Pi_{CD}(T)$ und $r_8 : \Pi_A(R) \bowtie \Pi_{CD}(T)$

(e) $r_9 : \sigma_{A>4 \wedge B=3} (\Pi_{AB}(R) \cup \Pi_{AB}(S))$ und $r_{10} : \sigma_{B=3} (\sigma_{A>4} (\Pi_{AB}(S) \cup \Pi_{AB}(R)))$

(f) $r_{11} : \sigma_{B \neq 3 \wedge B > 4} (\Pi_{BC}(T) \cup \sigma_{B \neq 3} (\Pi_{BC}(T)))$ und $r_{12} : \sigma_{B > 4} (\Pi_{BC}(T))$

Lösung:

(a) **Nein:** Der Join in r_1 degeneriert zu einem Kreuzprodukt. Dies ist nicht der Fall in r_2 .

(b) **Ja:** Die Projektion auf BC in r_3 ist unproblematisch, weil BC genau die gemeinsamen Attribute von S und T sind. Für den Semi-Join ändert sich daher nichts.

(c) **Nein:** Der Ausdruck r_6 ist strenger, da Tupel aus S nur erhalten bleiben wenn $B = 31$ gilt. In r_5 reicht es, wenn $B = 13 \vee B = 31$ erfüllt ist.

(d) **Ja:** Da die Argumente des Join-Operators in r_8 keine Attribute gemeinsam haben degeneriert der Join zu einem Kreuzprodukt.

(e) **Ja:** Die Konjunktion in der Selektionsbedingung kann aufgebrochen werden. Weiters gilt $X \cup Y = Y \cup X$ und $\varphi \wedge \psi \equiv \psi \wedge \varphi$.

- (f) **Ja:** Zunächst kann $B \neq 3 \wedge B > 4$ zu $B > 4$ vereinfacht werden. Außerdem kann die Vereinigung mit $\sigma_{B \neq 3}(\Pi_{BC}(T))$ weggelassen werden weil $\sigma_{B \neq 3}(\Pi_{BC}(T)) \subseteq \Pi_{BC}(T)$ gilt.

Aufgabe 6 (Größenabschätzung) [0.5 Punkte] Gegeben sind die Relationen $R(\underline{ABC})$ mit 4 Tupeln, $S(\underline{ACD})$ mit 6 Tupeln und $T(\underline{BCD})$ mit 8 Tupeln.

Geben Sie die minimale bzw. maximale Größe (= Anzahl der Tupel) der durch die folgenden Ausdrücke entstehenden Relationen an:

- (a) $S \bowtie T$ **Lösung:** (min 0/max 6)
- (b) $\Pi_{CD}(T) \cup \Pi_{CD}(\rho_{[D \leftarrow A]}(R))$ **Lösung:** (min 8/max 12)
- (c) $\Pi_A(S) \times \Pi_{CD}(T)$ **Lösung:** (min 8/max 48)

Aufgabe 7 (Formalisieren von Anfragen) [0.5 Punkte] Gegeben ist das folgende Relationenschema:

- S = Sammler(SNr, VorName, NachName, Alter, Beruf)
- C = CD(CDNr, Titel, Interpret, Laufzeit)
- L = LimitedCD(CDNr, Bezeichnung, Herausgeber)
- B = besitzt(SNr, CDNr)

Drücken Sie folgende Anfragen in relationaler Algebra aus.

Hinweis: Die Laufzeit wird in Sekunden gespeichert.

- (a) Wir suchen jene jungen Sammler (jünger als 19), die eine CD von "ACDC" besitzen, deren Laufzeit mehr als 50 Minuten beträgt.
- (b) Gesucht sind jene nicht limitierten CDs, die kein einziger Student (Hinweis: Attribut Beruf) besitzt.

Betrachten Sie nun den folgenden Ausdruck in relationaler Algebra. Beschreiben Sie in eigenen Worten, welches Ergebnis dieser Ausdruck liefert.

$$(c) \Pi_{\text{VorName, NachName}} \left[\left(S \bowtie (B \div \Pi_{\text{CDNr}}(\sigma_{\text{Interpret}='ACDC'}(C))) \right) - \sigma_{\text{Alter} \geq 30}(S) \right]$$

Lösung:

- (a) $\sigma_{\text{Alter} \leq 18}(S) \bowtie (B \bowtie \sigma_{\text{Interpret}='ACDC' \wedge \text{Laufzeit} > 3000}(C))$
- (b) $C \bowtie \left[\left(\Pi_{\text{CDNr}}(C) - \Pi_{\text{CDNr}}(L) \right) - \Pi_{\text{CDNr}} \left(B \bowtie \sigma_{\text{Beruf}='Student'}(S) \right) \right]$
 oder: $C \bowtie \left[\Pi_{\text{CDNr}}(C) - \left(\Pi_{\text{CDNr}}(L) \cup \Pi_{\text{CDNr}}(B \bowtie \sigma_{\text{Beruf}='Student'}(S)) \right) \right]$

- (c) Es werden die Namen jener fleißigen Sammler gesucht, die alle CDs von "ACDC" besitzen, jedoch jünger als 30 Jahre alt sind.

Aufgabe 8 (Relationenkalkül) [0.75 Punkte] Übersetzen Sie die Anfragen, die Sie in Aufgabe 7 (a) und Aufgabe 7 (b) in relationaler Algebra formuliert haben, in sichere Ausdrücke des Tupel- und Domainkalküls. Geben Sie bei jeder der Anfragen an, in welchem Kalkül sie formuliert ist.

Lösung:

- (a) Tupelkalkül: $\left\{ s \mid s \in S \wedge s.\text{Alter} \leq 18 \wedge \exists b \in B (s.\text{SNr} = b.\text{SNr} \wedge \exists c \in C (b.\text{CDNr} = c.\text{CDNr} \wedge c.\text{Interpret} = \text{'ACDC'} \wedge c.\text{Laufzeit} > 3000)) \right\}$

Domänenkalkül: $\left\{ [s, v, n, a, b] \mid [s, v, n, a, b] \in S \wedge a \leq 18 \wedge \exists c ([s, c] \in B \wedge \exists t, i, l ([c, t, i, l] \in C \wedge i = \text{'ACDC'} \wedge l > 3000)) \right\}$

- (b) Tupelkalkül: $\left\{ c \mid c \in C \wedge \neg \exists l \in L (c.\text{CDNr} = l.\text{CDNr}) \wedge \neg \exists b \in B (c.\text{CDNr} = b.\text{CDNr} \wedge \exists s \in S (b.\text{SNr} = s.\text{SNr} \wedge s.\text{Beruf} = \text{'Student'})) \right\}$

oder z.B.: $\left\{ c \mid c \in C \wedge \forall l \in L (c.\text{CDNr} \neq l.\text{CDNr}) \wedge \forall b \in B (c.\text{CDNr} \neq b.\text{CDNr} \vee \forall s \in S (b.\text{SNr} \neq s.\text{SNr} \vee s.\text{Beruf} \neq \text{'Student'})) \right\}$

Domänenkalkül: $\left\{ [c, t, i, l] \mid [c, t, i, l] \in C \wedge \neg \exists b, h ([c, b, h] \in L) \wedge \forall s ([s, c] \in B \Rightarrow (\forall v, n, a, b ([s, v, n, a, b] \in S \Rightarrow (b \neq \text{'Student'}))) \right\}$

oder z.B.: $\left\{ [c, t, i, l] \mid [c, t, i, l] \in C \wedge \forall b, h (\neg [c, b, h] \in L) \wedge \neg \exists s ([s, c] \in B \wedge (\exists v, n, a, b ([s, v, n, a, b] \in S \wedge (b = \text{'Student'}))) \right\}$

Gesamter Stoff

Aufgabe 9 (Falsche Merksätze) [0.5 Punkte] Ihr Kollege Karl Krocha erklärt einige wichtige Zusammenhänge, die er sich für die Prüfung merken will. Leider hat er manche Dinge falsch verstanden. Finden Sie die Fehler, begründen Sie warum manche Erklärungen falsch sind, und stellen Sie die falschen Aussagen richtig.

- (a) Wenn ich ein ER Diagramm in ein Relationenschema überführe muss ich pro Entität genau eine neue Relation erzeugen – ansonsten werden keine Relationen erzeugt.
- (b) Übrigens: Man kann die ternären Beziehungen ohne Informationsverlust immer durch binäre ersetzen, dann wird es gleich deutlich einfacher.

- (c) Im Grunde reichen die Basisoperatoren, um alle Operatoren in der relationalen Algebra auszudrücken. Außerdem lässt sich jeder Ausdruck in relationaler Algebra in SQL formulieren.
- (d) Bei den Schlüsseln von Relationen muss man wirklich aufpassen: Ein Schlüssel muss minimal sein – das heißt es darf keinen Schlüssel geben, der weniger Attribute umfasst. Es kann also zum Beispiel keinen Schlüssel mit drei Attributen geben, wenn es bereits einen Schlüssel mit zwei Attributen gibt.
- (e) Der Domänen- und Tupelkalkül sind sich recht ähnlich, allerdings wird beim Domänenkalkül über ganze Tupel quantifiziert.

Lösung:

- (a) **Falsch:** Für N:M (und fallweise 1:N) Beziehungen müssen natürlich weitere Relationen erzeugt werden.
- (b) **Falsch:** Dies stimmt im Allgemeinen nicht. Es kann zu Informationsverlust kommen, wenn ternäre Beziehungen durch binäre ersetzt werden.
- (c) **Richtig**
- (d) **Falsch:** Der Begriff “minimal” wurde missverständlich verwendet: Richtig ist Teilmengenminimalität! Es kann sehr wohl mehrere Schlüssel geben die unterschiedlich viele Attribute umfassen.
- (e) **Falsch:** Hier wurden Domänenkalkül und Tupelkalkül vertauscht. Im Tupelkalkül wird über ganze Tupel quantifiziert.