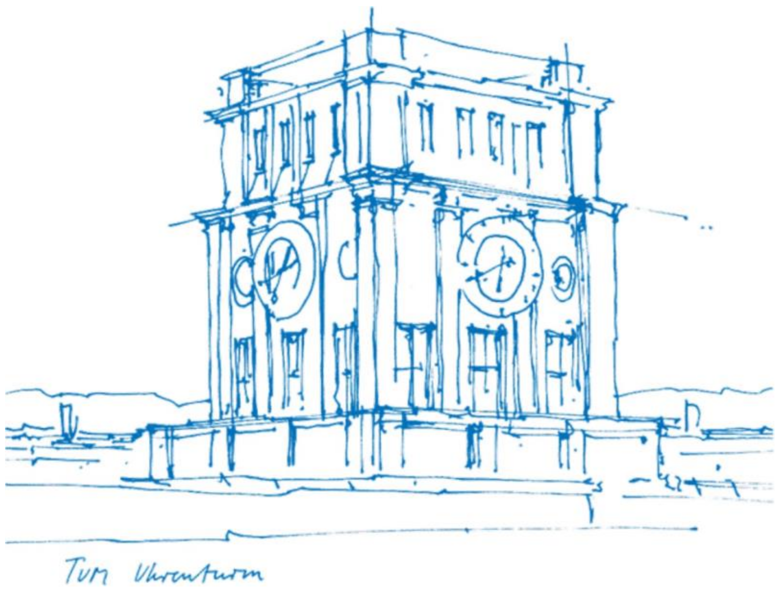
Disclaimer:
Die Slides der Tutorübung dienen nur zur Ergänzung und werden **nicht** fachlich geprüft!
Einige Übungen sind aus der Vorlesung “Grundlagen: Datenbanken” übernommen und dienen ebenfalls nur zur Ergänzung.

Grundlagen Datenbanken – Tutorium 02

Sebastian Reichbauer (ge64rom@mytum.de)



Agenda

- Wiederholung
- Aufgabe 1
- Aufgabe 2
- Aufgabe 3
- Aufgabe 4

Organisatorisches

Download der Folien:
home.in.tum.de/~reichbau
(alle Folien privat ohne vollständigen Stoff)

Hausaufgaben-Upload:
home.in.tum.de/~reichbau

Chair for Database Systems Department of
Informatics Technical University of Munich

Wiederholung

Relationale Algebra - Übersicht

Selektion σ
Projektion Π
Kreuzprodukt $\times$
Umbenennung ρ
Vereinigung $\cup$
Schnitt $\cap$
Differenz $\setminus$
Division $\div$

Natürlicher/Inner Join $\bowtie$
Left/Right Outer Join $⟕$/$⟖$
Full Outer Join $⟗$
Left/Right Semi Join $\ltimes$/$\rtimes$
Left/Right Anti Join $\triangleright$/$\triangleleft$

Und $\wedge, \vee, \neg, =, \neq, >, <, \geq, \leq$...

Relationen studenten & pruefen

Beispiel

matrnr	name	semester
24002	Xenokrates	18
25403	Jonas	12
26120	Fichte	10
26830	Aristoxenos	8
27550	Schopenhauer	6
28106	Carnap	3
29120	Theophrastos	2
29555	Feuerbach	2

matrnr	vorlnr	persnr	note
28106	5001	2126	1.0
25403	5041	2125	2.0
27550	4630	2137	2.0

→ Bilden Basis für folgende Ausdrücke

LEFT/RIGHT OUTER JOIN ⟕/⟖

Linke/rechte Relation ohne JOIN Partner?
→ Übernahme der Tupel e.g. mit **NULL-Einträgen**.

Beispiel: LEFT OUTER JOIN

L		
A	B	C
a_1	b_1	c_1
a_2	b_2	c_2

⟕

R		
C	D	E
c_1	d_1	e_1
c_3	d_2	e_2

=

Resultat				
A	B	C	D	E
a_1	b_1	c_1	d_1	e_1
a_2	b_2	c_2	$-$	$-$

FULL OUTER JOIN ⟗

Intuition: Rechter und linker OUTER JOIN **zusammen**.
⇒ Tupel ohne JOIN Partner werden trotzdem übernommen.

L

A	B	C
a_1	b_1	c_1
a_2	b_2	c_2

⟗

R

C	D	E
c_1	d_1	e_1
c_3	d_2	e_2

=

Resultat

A	B	C	D	E
a_1	b_1	c_1	d_1	e_1
a_2	b_2	c_2	–	–
–	–	c_3	d_2	e_2

Left/Right Semi JOIN ⋉/⋊

Semi JOINs können für **Existenzquantifizierung** eingesetzt werden.

Aussortierung der Tupel ohne JOIN Partner bzw. Beibehaltung der Tupel mit JOIN Partner.

Beispiel: Left Semi JOIN

$$\begin{array}{|c|c|c|}\hline \multicolumn{3}{|c|}{\textbf{L}} \\ \hline A & B & C \\ \hline a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ \hline \end{array} \ltimes \begin{array}{|c|c|c|}\hline \multicolumn{3}{|c|}{\textbf{R}} \\ \hline C & D & E \\ \hline c_1 & d_1 & e_1 \\ c_3 & d_2 & e_2 \\ \hline \end{array} = \begin{array}{|c|c|c|}\hline \multicolumn{3}{|c|}{\textbf{Resultat}} \\ \hline A & B & C \\ \hline a_1 & b_1 & c_1 \\ \hline \end{array}$$

Left/Right Anti JOIN $\triangleright$/ $\triangleleft$

Gegenteil vom Left/Right Semi JOIN (negierte Existenzquantifizierung)
→ Aussortierung der Tupel mit JOIN Partner bzw. Beibehaltung der Tupel ohne JOIN Partner.

Beispiel: Left Anti JOIN

L

A	B	C
a_1	b_1	c_1
a_2	b_2	c_2

$\triangleright$

R

C	D	E
c_1	d_1	e_1
c_3	d_2	e_2

$=$

Resultat

A	B	C
a_2	b_2	c_2

Mengenoperationen

Voraussetzung

Gleiche Attributnamen und -anzahl & gleiche Domänen

Prof1

PersNr	Name
1	Moerkotte
2	Kemper

Prof2

PersNr	Name
2	Kemper
3	Weikum

Vereinigung ∪

Prof1 ∪ Prof2

PersNr	Name
1	Moerkotte
2	Kemper
3	Weikum

Mengenoperationen

Voraussetzung

Gleiche Attributnamen und -anzahl & gleiche Domänen

Schnitt ∩

$$\mathsf{Prof1} \cap \mathsf{Prof2}$$

PersNr	Name
2	Kemper

Differenz /

$$\mathsf{Prof1} \setminus \mathsf{Prof2}$$

PersNr	Name
1	Moerkotte

Relationale Division R1 ÷ R2

Die relationale Division kann für **Allquantifizierung** eingesetzt werden.
Voraussetzung: R2 ⊆ R1 (siehe Vorlesung für formale Definition)

besucht

MatrNr	Nr
1	1
1	2
2	1
2	3
3	1
3	2
3	3

Vorl1

Nr
1
2

$\longrightarrow$

besucht $\div$ Vorl1

MatrNr
1
3

Intuitiv

Das Ergebnis enthält nur Attribute aus R_1 die nicht in R_2 enthalten sind. Dabei werden nur die Tupel aus R_1 übernommen, welche für **alle** Tupel in R_2 einen zugehörigen Eintrag besitzen.

Relationaler Tupelkalkül

Bindung der Variablen an Tupel der jeweiligen Relationen.

Syntax einer Anfrage:

$$\{t \mid P(t)\}$$

t (freie) Tupelvariable: **Ergebnisspezifikation**
P Prädikat / Formel: **Herleitung**

t freie Variable von P → t ist nicht durch ∃ oder ∀ quantifiziert.

Syntax zur Ausgabe neuer Tupel per Tupelkonstruktor [...]:
$\{[t1.A_1, ..., tn.A_n] \mid P(t_1, ..., t_n)\}$

Relationaler Tupelkalkül - Formal

Formeln werden aus *Atomen* folgender Form zusammengebaut:

- $s \in R$, mit s Tupelvariable und R Relationenname
- $s.A \, \phi \, t.B$, mit s und t Tupelvariablen, A und B Attributnamen und ϕ Vergleichsoperator ($=$, $\neq$, $\leq$, ...)
- $s.A \, \phi \, c$ mit c Konstante

Formeln werden nach folgenden Regeln aufgebaut:

- Alle Atome sind Formeln
- Ist P Formel, so auch $\neg P$ und (P)
- Sind P_1 und P_2 Formeln, so auch $P_1 \wedge P_2$, $P_1 \vee P_2$ und $P_1 \Rightarrow P_2$
- Ist $P(t)$ Formel mit freier Variable t, so auch

$$\forall t \in R(P(t)) \quad \text{und} \quad \exists t \in R(P(t))$$

Relationaler Tupelkalkül - Sicherheit

Problem: **Unendliches Ergebnis**

$$\{n \mid \neg(n \in Professoren)\}$$

Lösung: Einschränkung der Anfragen auf *sichere* Ausdrücke.

Ergebnis des Ausdrucks muss Teilmenge der *Domäne* der Formel sein. Diese umfasst

- alle in der Formel vorkommenden Konstanten sowie
- alle Attributwerte von Relationen, die in der Formel referenziert werden.

Anmerkung: Negierte Existenzquantoren zu verwenden ist potentiell also gefährlich.

Relationaler Tupelkalkül - Beispiel

*Finden sie die **MatrNr** aller **Studenten**, die **alle vierstündigen** Vorlesungen **gehört** haben.*

$$\{[s.MatrNr] \mid s \in Studenten \wedge \forall v \in Vorlesungen(v.SWS = 4 \Rightarrow \exists h \in hoeren(h.VorlNr = v.VorlNr \wedge h.MatrNr = s.MatrNr))\}$$

Relationaler Domänenkalkül

Bindung der Variablen an Domänen(**Wertemengen** von **Attributen**).

Syntax einer Anfrage:

$$\{[v_1, v_2, ..., v_n] \mid P(v_1, v_2, ..., v_n)\}$$

$v_1, v_2, ..., v_n$ (freie) Domänenvariablen: **Ergebnisspezifikation**
P Prädikat / Formel: **Herleitung**

Wichtig: Variablennamen eindeutig wählen!
$\rightarrow$ Joinbedingungen können implizit durch Verwendung derselben Variablen spezifiziert werden.
Ohne weitere Spezifikation: jew. Attributreihenfolge beachten!

$\exists\ v_1(\exists\ v_2(\exists\ v_3(P(v_1, v_2, v_3))))$ ist verkürzt $\exists\ v_1, v_2, v_3(P(v_1, v_2, v_3))$

Relationaler Domänenkalkül - Formal

Formeln werden aus Atomen folgender Form zusammengebaut:

- $[w_1, w_2, \ldots, w_m] \in R$, mit m-stelliger Relation R und Domänenvariablen $w_1, \ldots, w_m$
- $x \, \phi \, y$, mit x und y Domänenvariablen, ϕ Vergleichsoperator
- $x \, \phi \, c$, mit Konstante c

Formeln werden nach folgenden Regeln aufgebaut:

- Alle Atome sind Formeln
- Ist P Formel, so auch $\neg P$ und (P)
- Sind P_1 und P_2 Formeln, so auch $P_1 \vee P_2$, $P_1 \wedge P_2$ und $P_1 \Rightarrow P_2$
- Ist $P(v)$ Formel mit freier Variable v, so auch $\exists v(P(v))$ und $\forall v(P(v))$

Relationaler Domänenkalkül - Sicherheit

Problem: **Unendliches Ergebnis**

$$\{[p,n,r,o] \mid \neg([p,n,r,o] \in Professoren)\}$$

Lösung: Einschränkung der Anfragen auf *sichere* Ausdrücke:

- Falls Tupel $[c_1, c_2, \ldots, c_n]$ im Ergebnis, so muss jedes $c_i (1 \leq i \leq n)$ in der Domäne von P enthalten sein. $\rightarrow$ Kein Unterschied zu Tupelkalkül
- Für jede Teilformel $\exists x (P_1(x))$ muss gelten, dass P_1 nur für Elemente aus der Domäne von P_1 erfüllbar ist - oder evtl. für gar keine.
- Für jede Teilformel $\forall x (P_1(x))$ muss gelten, dass sie erfüllt ist, gdw. $P_1(x)$ für alle Werte der Domäne von P_1 erfüllt ist. Für alle anderen Werte muss sie sowieso erfüllt sein.

Relationaler Domänenkalkül - Beispiel

Finden sie die ***MatrNr*** *aller* ***Studenten****, die* ***alle vierstündigen*** *Vorlesungen* ***gehört*** *haben.*

$$\{[m] \mid \exists n,s([m,n,s] \in Studenten \land$$
$$\forall v,t,sws,g([v,t,sws,g] \in Vorlesungen \land sws = 4$$
$$\Rightarrow [m,v] \in hoeren))\}$$

Blatt 4 – Aufgabe 1

Aufgabenstellung – Aufgabe 1

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema im Tupel- und Domänenkalkül:

a) Finden Sie die Vorlesungen, die keine Hörer haben.

b) Finden Sie die Studenten, die alle Vorlesungen hören.

Lösung – Aufgabe 1

a)

Formulierung im Tupelkalkül

$$\{\mathrm{v} \mid \mathrm{v} \in \mathrm{Vorlesungen} \land \nexists\, \mathrm{h} \in \mathrm{h\ddot{o}ren}(\mathrm{v.VorlNr} = \mathrm{h.VorlNr})\}$$

oder

$$\{\mathrm{v} \mid \mathrm{v} \in \mathrm{Vorlesungen} \land \forall\, \mathrm{h} \in \mathrm{h\ddot{o}ren}(\mathrm{v.VorlNr} \neq \mathrm{h.VorlNr})\}$$

Formulierung im Domänenkalkül

$$\{[\mathrm{v,t,s,g}] \mid [\mathrm{v,t,s,g}] \in \mathrm{Vorlesungen} \land \nexists\, \mathrm{m}([\mathrm{m,v}] \in \mathrm{h\ddot{o}ren})\}$$

Lösung – Aufgabe 1

b)

Formulierung im Tupelkalkül

$$\{s \mid s \in \text{Studenten} \;\wedge\; \forall\, v \in \text{Vorlesungen}(\\ \exists\, h \in \text{hoeren}(v.\text{VorlNr} = h.\text{VorlNr} \wedge s.\text{MatrNr} = h.\text{MatrNr}))\}$$

Formulierung im Domänenkalkül

$$\{[m,n,s] \mid [m,n,s] \in \text{Studenten} \;\wedge\\ \forall\, v,t,sws,g([v,t,sws,g] \in \text{Vorlesungen} \Rightarrow [m,v] \in \text{hören})\}$$

Blatt 4 – Aufgabe 2

Chair for Database Systems Department of
Informatics Technical University of Munich

Aufgabenstellung – Aufgabe 2

Gegeben seien die beiden Relationen $R:\{[a_1, \ldots, a_n]\}$ und $S:\{[b_1, \ldots, b_m]\}$. Geben Sie die folgenden Ausdrücke im Tupel- und Domänenkalkül an:

a) $Q1 := R \bowtie_{a1=b1} S$

b) $Q2 := R ⟕_{a1=b1} S$

c) $Q3 := R \rtimes_{a1=b1} S$

d) $Q4 := R \triangleleft_{a1=b1} S$

Lösung – Aufgabe 2

a) *Q1 := R* $\bowtie_{a1=b1}$ *S*

Tupelkalkül

$$Q_1 := \{[r.a_1, \ldots, r.a_n, s.b_1, \ldots, s.b_m] \mid r \in R \ \wedge \ s \in S \ \wedge \ r.a_1 = s.b_1\}$$

Domänenkalkül

$$Q_1 := \{[a_1, \ldots, a_n, b_1, \ldots, b_m] \mid [a_1, \ldots, a_n] \in R \wedge [b_1, \ldots, b_m] \in S \wedge a_1 = b_1\}$$

Lösung – Aufgabe 2

b) *Q2 := R* ⟕$_{a1=b1}$ *S*

Tupelkalkül

$$Q_2 := Q_1 \cup \{[r.a_1, \ldots, r.a_n, b_1 : \text{null}, \ldots, b_m : \text{null}] \mid r \in R \land \nexists\, s \in S(r.a_1 = s.b_1)\}$$

Domänenkalkül

$$Q_2 := Q_1 \cup \{[a_1, \ldots, a_n, b_1 : \text{null}, \ldots, b_m : \text{null}] \mid [a_1, \ldots, a_n] \in R \land \nexists\, c_2, \ldots, c_m([a_1, c_2, \ldots, c_m] \in S)\}$$

Lösung – Aufgabe 2

c) $Q3 := R \rtimes_{a1=b1} S$

Tupelkalkül

$$Q_3 := \{s \mid s \in S \land \exists\, r \in R(r.a_1 = s.b_1)\}$$

Domänenkalkül

$$Q_3 := \{[b_1, \ldots, b_m] \mid [b_1, \ldots, b_m] \in S \land \exists\, a_2, \ldots, a_n([b_1, a_2, \ldots, a_n] \in R)\}$$

Lösung – Aufgabe 2

d) $Q4 := R \lhd_{a1=b1} S$

Tupelkalkül

$$Q_4 := \{s \mid \; s \in S \wedge \nexists\, r \in R(r.a_1 = s.b_1)\}$$

Domänenkalkül

$$Q_4 := \{[b_1, \ldots, b_m] \mid [b_1, \ldots, b_m] \in S \wedge \nexists\, a_2, \ldots, a_n([b_1, a_2, \ldots, a_n] \in R)\}$$

Chair for Database Systems Department of
Informatics Technical University of Munich

Blatt 4 – Aufgabe 3

Aufgabenstellung – Aufgabe 3

Formulieren Sie folgende Anfragen auf dem bekannten Universitätsschema in der Relationenalgebra:

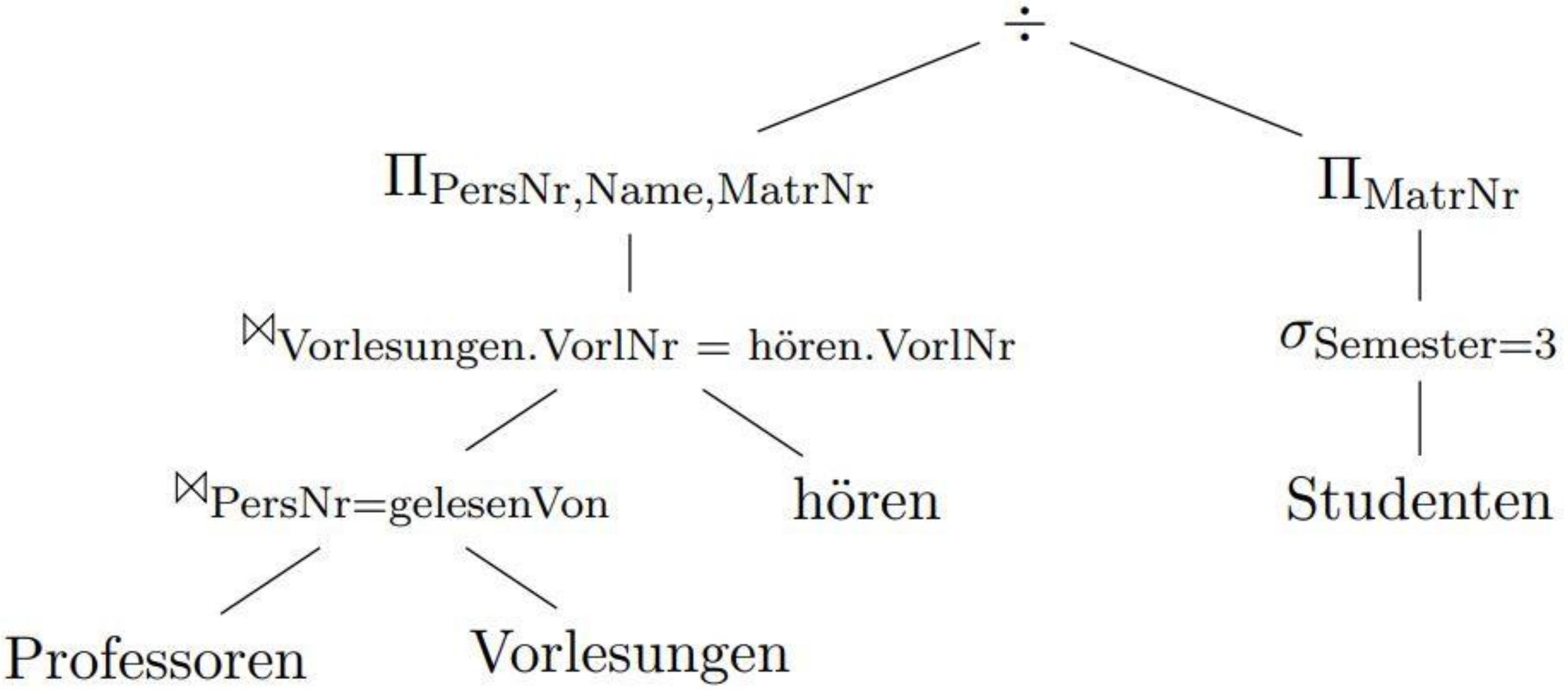
a) Finden Sie alle bei den Drittsemestern beliebte Professoren. Ein Professor ist bei einem gegebenen Semester beliebt, wenn alle Studenten aus diesem Semester mindestens eine seiner Vorlesungen hören (aber nicht notwendigerweise alle dieselbe).

b) Finden Sie alle Grundlagenvorlesungen. Eine Grundlagenvorlesung ist eine Vorlesung, die keine Voraussetzungen hat.

c) Carnap will eine Seminararbeit einreichen. Er will in seiner Danksagung alle Professoren und ihre Assistenten erwähnen, deren Vorlesungen er hört. Geben Sie eine Anfrage an, die alle diese Namen ermittelt.

Lösung – Aufgabe 3

a) Um Anfragen mit Allquantifizierung auszudrücken, kann man in der relationalen Algebra den Divisionsoperator verwenden. Wichtig ist hier, dass das Schema des Divisors eine Teilmenge des Schemas des Dividenden ist. Daher projizieren wir vor der Division den Dividenden auf die Matrikelnummer.



Lösung – Aufgabe 3

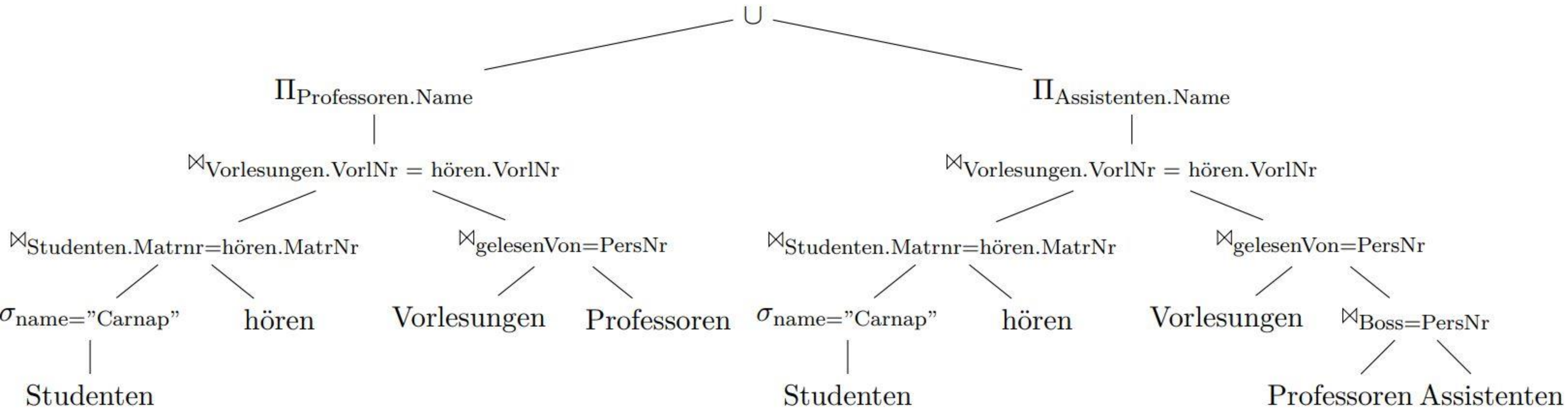
b) Um Anfragen der Art "Finde alle Elemente für die es kein Pendant (Joinpartner) gibt" auszudrücken, verwendet man den Anti-Semi-Join.

$$\triangleright_{\text{VorlNr} = \text{Nachfolger}}$$

Vorlesungen   voraussetzen

Lösung – Aufgabe 3

c) Wir suchen zunächst alle Professoren, deren Vorlesung Carnap besucht und dann (in einer beinahe identischen Anfrage) alle ihre Assistenten. Um die Namensliste zu bekommen, bilden wir dann die Vereinigungsmenge von beiden. Wichtig ist hier, jeweils auf den Namen zu projizieren, da für die Vereinigung beide Schemata gleich sein müs-



Blatt 4 – Aufgabe 4

Aufgabenstellung – Aufgabe 4

Gegeben sei die folgende Relation `Zehnkampf` mit Athletennamen und den von ihnen erreichten Punkten im Zehnkampf:

Name	Punkte
Eaton	8869
Suarez	8523
Behrenbruch	8126
Hardee	8671
...	...

a) Ermitteln Sie die Goldmedaillengewinner in relationaler Algebra. Eine Goldmedaille bekommen alle, für die gilt: es gibt niemand besseren (also mit mehr Punkten).

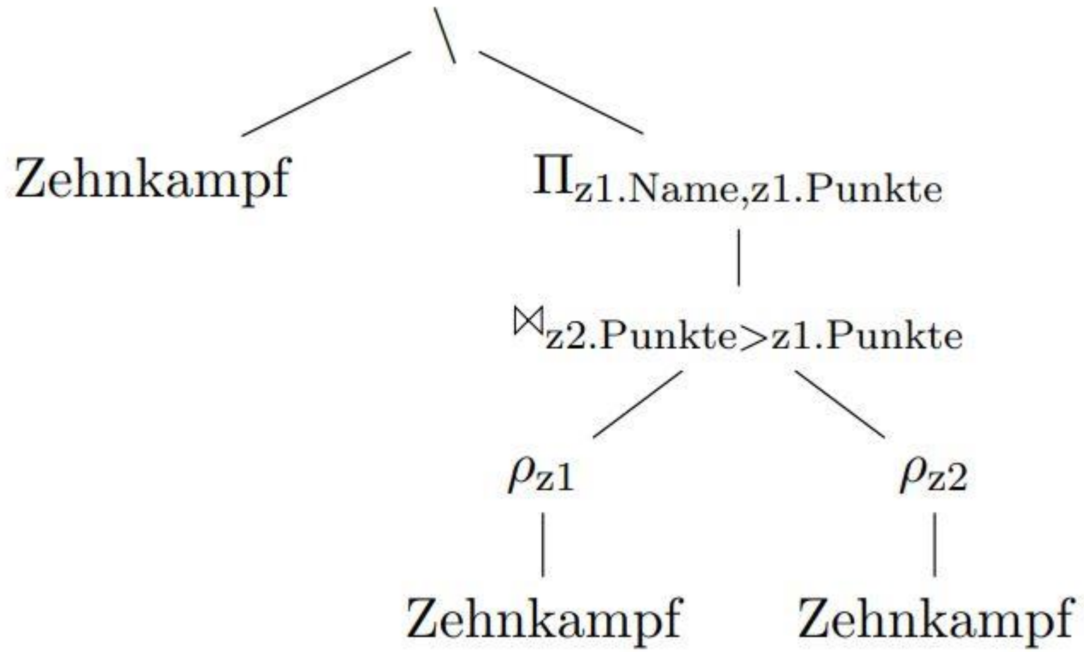
b) Ermitteln Sie die Silbermedaillengewinner im Tupelkalkül. Eine Silbermedaille bekommen alle, für die gilt: es gibt genau eine/n bessere/n.

Lösung – Aufgabe 4

a) Goldmedaillengewinner in relationaler Algebra:

$$(\text{Zehnkampf} \setminus (\Pi_{\text{Name,Punkte}}(\rho_{\text{z1}}\text{Zehnkampf} \bowtie_{\text{z2.Punkte>Punkte}} \rho_{\text{z2}}\text{Zehnkampf})))$$

In Operatorbaumdarstellung:



Lösung – Aufgabe 4

b) Silbermedaillengewinner im Tupelkalkül:

$$
\begin{aligned}
\{k| \quad & k \in \text{Zehnkampf} \wedge \\
& \exists k_{gold} \in \text{Zehnkampf}(\\
& k_{gold}.\text{Punkte} > k.\text{Punkte} \wedge \forall k_{andere} \in \text{Zehnkampf}(\\
& k_{andere}.\text{Punkte} \geq k_{gold}.\text{Punkte} \Rightarrow k_{andere}.\text{Name} = k_{gold}.\text{Name}) \wedge \\
& \neg\exists k_{zwischen} \in \text{Zehnkampf}(k_{zwischen}.\text{Punkte} > k.\text{Punkte} \wedge \\
& k_{zwischen}.\text{Punkte} < k_{gold}.\text{Punkte}))\}
\end{aligned}
$$

Eine wunderschöne Woche noch!
Bis nächstes Mal!