

Pflichtmodul Informationssysteme (SS 2020)

Prof. Dr. Jens Teubner

Leitung der Übungen: Thomas Lindemann, Christoph Stahl

Übungsblatt Nr. 12

Ausgabe: 08.07.2019

Abgabe: —

Dieses Übungsblatt bildet eine Sammlung von Aufgaben, die teilweise Aufgaben in vergangenen Klausuren ähnlich sind. Es dient zur Selbstkontrolle und soll einen Überblick über einiger der über das Semester hinweg behandelten Themen verschaffen. Zudem wurde jede Aufgabe mit Punkten annotiert. Diese geben an, wieviel Zeit für die Bearbeitung der Aufgabe vorgesehen wird. Ein Punkt entspricht einer Minute Bearbeitungszeit. Insgesamt gibt es 60 Punkte zu verteilen, für das Blatt sind somit 60 Minuten vorgesehen.

Der Stoff dieses Übungsblattes umfasst zwar ein großen Teil des Stoffes der Veranstaltung, soll aber keineswegs als vollständige Liste an klausurrelevanten Themen verstanden werden.

Aufgabe 1 (ER-Modellierung (*Entity Relation Models*) (3 + 4 + 3 Punkte))

In einem Schulungszentrum werden verschiedene Kurse angeboten, wobei ausgewählte Kurse den Inhalt anderer Kurse voraussetzen. Je Kurs gibt mindestens eine Lehrkraft, die auch verschiedene Kurse betreuen kann.

1. Erstellen Sie für diesen Diskursbereich ein ER-Diagramm. Verwenden Sie für die Funktionalität der Beziehungstypen die Min-/Max-Notation. (3 Punkte)

Aufgabe 2 (Relationen-Algebra (*Relational Algebra*) (6 + 8 Punkte))

1. Gegeben sei folgende Datenbank.

| <i>R</i> | | | <i>S</i> | | | <i>T</i> | | |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| <i>A</i> | <i>B</i> | <i>C</i> | <i>A</i> | <i>B</i> | <i>C</i> | <i>A</i> | <i>B</i> | <i>C</i> |
| 1 | 1 | x | 1 | 3 | x | 1 | 3 | y |
| 1 | 4 | y | 1 | 1 | x | 4 | 3 | y |
| 5 | 1 | x | 4 | 3 | y | 1 | 1 | x |

Werten Sie folgende Anfragen in relationaler Algebra aus. (3 + 3 Punkte)

(a) $R \div \pi_{B \leftarrow A, C}(S)$

(b) $(S - R) \bowtie T$

2. Der folgende Ausschnitt eines Datenbankschemas modelliert eine Datenbank an Schachspielenden. Die Relation *Player* modelliert einen Schachspielenden, die Relation *Game* ein Spiel zwischen zwei Schachspielenden. Das Attribut *Remis* gibt an, ob es sich bei dem Spiel um ein Unentschieden handelt, es kann *true* oder *false* sein.

$\text{sch}(\text{Player}) = (\underline{\text{PID}}, \text{Vorname}, \text{Nachname}, \text{Geburtsdatum})$

$\text{sch}(\text{Game}) = (\underline{\text{GID}}, \text{SiegerID}, \text{VerliererID}, \text{Remis})$

Hierbei ist *SiegerID* und *VerliererID* ein Fremdschlüssel für *Player*.

Geben Sie die folgenden natürlichsprachlichen Anfragen als Anfragen der relationalen Algebra an. (2 + 2 + 4 Punkte)

- (a) Welche Schachspielenden haben mindestens einmal gewonnen?. (Ausgabe: *Vorname*, *Nachname*)
- (b) Welche *verschiedenen* Schachspielende wurden am selben Tag geboren? (Ausgabe: *PID₁*, *PID₂*)
- (c) Welche Schachspielende haben in mindestens zwei Spielen gewonnen? (Ausgabe: *SiegerID*)

Aufgabe 3 (Tupel-Relationen-Kalkül (*Tuple Relational Calculus*) (3 + 3 + 3 Pt))

Übersetzen Sie die folgenden Ausdrücke der relationalen Algebra mit Hilfe der gegebenen Übersetzungsregeln im Anhang des Übungsblattes in äquivalente Ausdrücke des Safe TRC. Dabei soll jeder einzelne Übersetzungsschritt klar erkennbar sein.

1. $\sigma_{A=5}(R)$ für $\text{sch}(R) = (A, B, C)$

2. $\pi_{A,B}(R \cup S)$ für $\text{sch}(R) = (A, B, C)$ und $\text{sch}(S) = (A, B, C)$

3. $R \bowtie S$ für $\text{sch}(R) = (A, B, C)$ und $\text{sch}(S) = (A, D, E)$

Aufgabe 4 (SQL (*Structured Query Language*) (5 + 5 + 7 Punkte))

1. Im Folgenden werden Ihnen drei Momentaufnahmen von Datenbanken gegeben. Geben Sie an, wie Sie mit möglichst wenigen und möglichst kurzen SQL-Statements diese in einander überführen können. Hierbei stellt eine Tabelle mit hellgrauem Kopf eine View dar. (3 + 2 Punkte)

Von:

| <i>R</i> | | | <i>S</i> | | | <i>T</i> | | |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|
| <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>E</i> | <i>F</i> | <i>G</i> | <i>H</i> | <i>I</i> |
| 1 | 1 | x | 1 | 3 | x | 1 | 3 | y |
| 1 | 4 | y | 1 | 1 | x | 4 | 3 | y |
| 5 | 1 | x | 4 | 3 | y | 1 | 1 | x |

Über:

| <i>R</i> | | | <i>S</i> | | | <i>T</i> | | | <i>View</i> | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|-------------|----------|----------|----------|----------|----------|----------|
| <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>E</i> | <i>F</i> | <i>G</i> | <i>H</i> | <i>I</i> | <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>E</i> | <i>G</i> | <i>H</i> |
| 1 | 1 | x | 1 | 3 | x | 1 | 3 | y | 1 | 1 | x | 1 | 1 | 1 | 1 |
| 1 | 4 | y | 1 | 1 | x | 4 | 3 | y | 1 | 1 | x | 1 | 3 | 1 | 1 |
| 5 | 1 | x | 4 | 3 | y | 1 | 1 | x | 1 | 4 | y | 4 | 3 | 1 | 3 |
| | | | | | | | | | 1 | 4 | y | 4 | 3 | 4 | 3 |
| | | | | | | | | | 5 | 1 | x | 1 | 1 | 1 | 1 |
| | | | | | | | | | 5 | 1 | x | 1 | 3 | 1 | 1 |

Nach:

| <i>R</i> | | | <i>S</i> | | | <i>T</i> | | | <i>View</i> | | | | | | |
|----------|----------|----------|----------|----------|----------|----------|----------|----------|-------------|----------|----------|----------|----------|----------|----------|
| <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>E</i> | <i>F</i> | <i>G</i> | <i>H</i> | <i>I</i> | <i>A</i> | <i>B</i> | <i>C</i> | <i>D</i> | <i>E</i> | <i>G</i> | <i>H</i> |
| 1 | 1 | x | 1 | 3 | x | 1 | 3 | y | 1 | 1 | x | 1 | 3 | 1 | 1 |
| 1 | 4 | y | 4 | 3 | y | 1 | 1 | x | 1 | 4 | y | 4 | 3 | 1 | 3 |

Von → Über:**Über → Nach:**

2. Betrachten Sie den Semijoin Operator \bowtie . Gehen Sie davon aus, dass R und S die folgenden Schemata haben.

- $\text{sch}(R) = \{A, B, C\}$
- $\text{sch}(S) = \{A, D, E\}$

(a) Geben Sie $R \bowtie S$ nur mit Basisoperatoren der relationalen Algebra an. (3 Punkte)

(b) Geben Sie ein zu $R \bowtie S$ äquivalentes SQL-Statement an. (2 Punkte)

Hinweis: Es gibt kein Keyword `SEMI JOIN` in SQL.

3. Der folgende Ausschnitt eines Datenbankschemas modelliert eine Datenbank an Schachspielenden. Die Relation *Player* modelliert einen Schachspielenden, die Relation *Game* ein Spiel zwischen zwei Schachspielenden. Das Attribut *Remis* gibt an, ob es sich bei dem Spiel um ein Unentschieden handelt, es kann *true* oder *false* sein.

$\text{sch}(\text{Player}) = (\underline{\text{PID}}, \text{Vorname}, \text{Nachname}, \text{Geburtsdatum})$

$\text{sch}(\text{Game}) = (\underline{\text{GID}}, \text{SiegerID}, \text{VerliererID}, \text{Remis})$

Geben Sie die folgenden natürlichsprachlichen Anfragen als SQL-Anfragen an.

- (a) Wieviel Spiele hat *Garri Kasparow* gewonnen? (Ausgabe: *Anzahl*) (2 Punkte)

- (b) Welche Schachspielende haben noch nie gewonnen? (Ausgabe: *Vorname, Nachname*) (2 Punkte)

- (c) Wer ist der jüngste Spielende? (Ausgabe: *Vorname, Nachname*) (3 Punkte)
Hinweis: Verwenden Sie hier nicht **LIMIT**.

Aufgabe 5 (Schemanormalisierung (*Normalization*) (4 + 6 Punkte))

Gegeben seien das Relationenschema

$$\text{sch}(R) = DBINFO$$

sowie die Menge

$$\mathcal{F} = \{DB \rightarrow INFO, \quad I \rightarrow NFO, \quad DI \rightarrow B, \quad F \rightarrow O, \quad N \rightarrow D\}$$

von funktionalen Abhängigkeiten.

1. Zeigen Sie, dass das gegebene Schema von R mit den gegebenen funktionalen Abhängigkeiten \mathcal{F} **die Boyce-Codd Normalform verletzt**. (4 Punkte)

2. **Zerlegen Sie R mit Hilfe des BCNF-Zerlegungsalgorithmus aus der Vorlesung, sodass alle resultierenden Relationen in BCNF vorliegen.** (6 Punkte)

Geben Sie dabei jeweils an, entlang welcher funktionalen Abhängigkeit Sie das Schema zerlegen. Machen Sie nach Ablauf des Algorithmus die Schemata aller erzeugten Tabellen, die zu dem zerlegten Relationenschema gehören und alle zugehörigen funktionalen Abhängigkeiten kenntlich.

(mehr Platz für Ihre Lösung auf der nächsten Seite)

Anhang (*Appendix*)

In Vorlesung und Übung wurde diskutiert, wie ein beliebiger Ausdruck der relationalen Algebra in einen äquivalenten Ausdruck des Safe TRC übersetzt werden kann:

$$\text{TRC}(Exp) := \{v \mid \mathbb{T}(v, Exp)\} .$$

$\mathbb{T}(v, Exp)$ erzeugt dabei eine Formel, indem ein beliebiger Algebraausdruck Exp entlang seiner Struktur gemäß der folgenden Regeln zerlegt wird:

$$\mathbb{T}(v, R) := v \in R \quad (\text{wobei } R \text{ eine Basisrelation ist})$$

$$\mathbb{T}(v, \sigma_p(Exp)) := \mathbb{T}(v, Exp) \wedge p(v)$$

$$\mathbb{T}(v, \pi_{\langle A_1, \dots, A_k \rangle}(Exp)) := \exists u : \mathbb{T}(u, Exp) \wedge v \leftarrow \langle u.A_1, \dots, u.A_k \rangle$$

$$\mathbb{T}(v, Exp_1 \times Exp_2) := \exists u : \mathbb{T}(u, Exp_1) \wedge \exists w : \mathbb{T}(w, Exp_2) \wedge v \leftarrow \langle u, w \rangle$$

$$\mathbb{T}(v, Exp_1 \cup Exp_2) := \mathbb{T}(v, Exp_1) \vee \mathbb{T}(v, Exp_2)$$

$$\mathbb{T}(v, Exp_1 - Exp_2) := \mathbb{T}(v, Exp_1) \wedge \neg \mathbb{T}(v, Exp_2)$$