



## Exercise Sheet 6: Relational Algebra (until Thursday, 01.12.2016) (30 points)

**Please note:** you need **50%** of all exercise points to receive the *Studienleistung* for this lecture. In order to pass the RDB I Module, you need both the *Studienleistung* **and** you need to pass the exam. Exercises have to be turned in until **Thursday before the lecture** either in the lecture hall or into our mailbox at the IFIS floor (Mühlenpfordtstraße 23, 2<sup>nd</sup> floor). Please do not forget your **Matrike-Inummer** and your **tutorial group number** on your solutions. **If you forget** to write your Matrike-Inummer and/or your tutorial group number, you get **automatically 0 points**. Your solutions may be in German or English. Unless otherwise specified: **Always use your own words!**

### Exercise 6.1 (6 points)

Please answer the following questions

- 1) How are null-values treated, when applying an aggregation? (1 point)
- 2) How are duplicates treated, when applying an aggregation? (1 point)
- 3) What property must be given by two relations  $A$  and  $B$  so that set operations like  $\cup$ ,  $\cap$  or  $\setminus$  can be applied? (1 point)
- 4) How can the expression  $R \bowtie_{\theta} S$  be expressed with basic relational algebra operators? (1 point)
- 5) Briefly explain what the following *extended* relational algebra operators do: (2 points)
  - a. Left semi join:  $\ltimes$
  - b. Right outer join:  $\ltimes^r$

### Exercise 6.2 (1 point)

Another interesting relational algebra operator that was not covered in the lecture is the *anti-join*. The expression “ $R$  anti-join  $S$ ” returns all tuples in the results of expression  $R$  such that there are **no** tuples in the results of  $S$  with matching values for the shared attributes (the **opposite** idea of a semi-join). For example, the query “ $\text{Student}(\text{ID}, \text{name})$  *antijoin*  $\text{Enrolled}(\text{ID}, \text{course})$ ” returns the ID and name of all students who are not enrolled in any courses. Provide a relational algebra expression for the anti-join operator using the relations  $R$  and  $S$ .

### Exercise 6.3 (15 points)

Consider the following schema:

$\text{Student}(\underline{\text{name}}, \text{age}, \text{gender}, \text{studium})$

$\text{Eats\_at}(\underline{\text{name}}, \underline{\text{restaurant}})$

$\text{Eats}(\underline{\text{name}}, \underline{\text{dish}})$

$\text{Serves}(\underline{\text{restaurant}}, \underline{\text{dish}}, \text{price}, \text{type})$

In the relation  $\text{Serves}$ ,  $\text{type}$  refers to one of the following: “meat”, “turkey”, “chicken”, “seafood”

Write relational algebra expression for the following queries.

1. Show all the restaurants frequented by at least one student who is at least 19 years old and studies "Informatik". (2 points)
2. Show the names of all female students who eat seafood or turkey (or both) (2 points)
3. Show the names of all male students who eat "curry wurst" (2 points)
4. Show the name of the youngest male student who is studying "Psychology" (2 points)
5. Show on price average of the dishes of type "meat" (2 points)
6. Show all the restaurants that serve at least one dish that Sophie eats for more than € 16.00 (2 points)
7. Show the restaurant that serves the cheapest "Linsensuppe". (3 points)

### Exercise 6.4 (8 points)

Recall the *Student*, *Course* and *exam* tables from the *Query Optimization* detour of lecture 6:

Course

crs_no	title
41	Cooking rabbits
40	Destroying rings
42	Flying eagles

Student

mat_no	firstname	lastname	size
3519	Bilbo	Baggins	103
1473	Samwise	Gamgee	114
2308	Meriadoc	Brandybuck	135
1337	Erna	Broosh	86
2158	Frodo	Baggins	111
1104	Peregrin	Took	142
2480	Sméagol	NULL	98

exam

student	course	result
1473	41	1.0
3519	40	3.3
2480	40	1.7
1337	42	1.3
2480	41	4.0
2158	40	2.3

Based on these tables, create a relational algebra expression for each of the following queries:

- a) Return the titles of all courses Sméagol took an exam in. (2 points)
- b) Return the average size of all students that took no exam. (3 points)
- c) Return the first names of all students that have written the most exams. (3 points)