Dr. Stefan Brass
School of Information Sciences
University of Pittsburgh

April 26, 2000

INFSCI 2710 "Database Management"
— Solutions to Example Final Exam III —

Exercise 1 (SQL Errors) 6 Points

a)
```
SELECT ID, TITLE
FROM   FILM
WHERE  ID NOT IN(SELECT C.ID FROM CRITIQUE C)
```

☒ Correct

A common mistake was to mark this as incorrect because no tuple variable is declared for `FILM` and `ID` is used in the outer query without tuple variable. However, it is legal to leave out the tuple variable name in the `FROM` clause, then the table name `FILM` is used as tuple variable name, too. Also, attribute references need only a tuple variable when they would otherwise be ambiguous. But note that the tuple variable `C` declared in the subquery is not available in the outer query (this is like Pascal's block structure). Therefore, `ID` in the outer query can only refer to `FILM`.

b)
```
SELECT F.ID, F.TITLE
FROM   FILM F, CRITIQUE C
WHERE  F.ID = C.ID AND C.SOURCE IS NULL
```

☒ Wrong, Reason: If there is no critique/review, there will be no join partner. `C.SOURCE` can never be null, since it is part of the key. An outer join would work here.

c)
```
SELECT F.ID, F.TITLE
FROM   FILM F
WHERE  NOT EXISTS(SELECT * FROM CRITIQUE C)
```

☒ Wrong, Reason: This is an uncorrelated subquery, which under `NOT EXISTS` is almost always wrong. Here, if there is at least one `CRITIQUE`, no matter for what film, the output of the query will be empty.

d)
```
SELECT F.ID, F.TITLE
FROM   FILM F
WHERE  F.ID NOT IN (SELECT C.ID FROM CRITIQUE C
                    WHERE GRADE <> 'A')
```

☒ Wrong, Reason: This checks that the film has no reviews with other grades than “A”, but it also prints films without any reviews. It was required that there is at least one review with grade “A”.

e)
```
SELECT F.ID, F.TITLE
FROM   FILM F, CRITIQUE C
WHERE  F.ID = C.ID
AND    C.GRADE = 'A'
```

☒ Wrong, Reason: There could be one review with an “A” grade and another one with a “B” grade. This query would print the film, although it has not only ‘A” grades.

f)
```
SELECT F.ID, F.TITLE
FROM   FILM F, CRITIQUE C
WHERE  F.ID = C.ID AND C.GRADE = 'A'
AND    NOT EXISTS(SELECT * FROM CRITIQUE X
                  WHERE X.ID = F.ID AND X.GRADE <> 'A')
```

☒ Correct

Exercise 2 (FDs, BCNF) **7 Points**

a) What does the functional dependency "`QUESTION, ANSWER` → `CORRECT`" mean? Please check the right explanation (only one is correct):

☒ The correctness of an answer to a question is independent from the student.

Many students thought that the first answer "Every question has only one correct answer" were correct. However, the following example relation satisfies the FD "`QUESTION, ANSWER` → `CORRECT`", but has two correct answers for one question:

MULTIPLE_CHOICE_TEST			
STUDENT	QUESTION	ANSWER	CORRECT
John Smith	a	1	Y
Maria Brown	a	2	Y

The FD is only equivalent to the statement/constraint "Every question has only one correct answer" if for all database states, either both are true, or both are false.

b) We want to make sure that not more than one answer to a query can be correct. Would the FD "`QUESTION, CORRECT` → `ANSWER`" do the job?

☒ This FD is not even satisfied in the given table, but the table satisfies "not more than one correct answer". The FD is too strong (although it would imply that every question has only one correct answer).

For `QUESTION='b'` and `CORRECT='N'`, the answer is not uniquely determined in the given example relation.

c) Can functional dependencies be determined by looking at example data?

☒ Example data can only show that functional dependencies do not hold. You cannot conclude from example data that an FD must hold in general.

The following example shows that the last answer is not correct:

COURSE		
CRN	TITLE	INSTRUCTOR
11111	Database Management	Brass
22222	Database Management	Brass
33333	Document Processing	Spring
44444	Document Processing	Mundie

Here the FD "`CRN` → `TITLE, INSTRUCTOR`" holds (so CRN is a key, some people would say because CRN is a key), but also "`INSTRUCTOR` → `TITLE`" holds. But `INSTRUCTOR` is not a key, since "`INSTRUCTOR` → `CRN`" does not hold (because of the two sessions of database management). With respect to these FDs, it would violate BCNF. But when we add the course "Client-Server" taught by Michael Spring, the FD "`INSTRUCTOR` → `TITLE`" is violated. So it does not hold in general. Without this FD, the relation is in BCNF, and indeed, the relation is a good design. So the BCNF test is not based on particular database states, but on constraints (FDs), which must hold in general.

d) Is the FD "`STUDENT, QUESTION` → `CORRECT`" implied by the given two FDs?

☒ Yes.

e) What is the (minimal) key of the relation with respect to the two given FDs? (It has only one key.)

☒ `STUDENT, QUESTION`

f) Is the relation in BCNF?

☒ No. The FD "`QUESTION, ANSWER` → `CORRECT`" violates BCNF.

g) What would be a lossless decomposition (only one answer is correct)?

☒ `R1(STUDENT, QUESTION, ANSWER)` and
`R2(QUESTION, ANSWER, CORRECT)`