Uppsala University
Department of Information Technology

Exam 2013-10-24

Database Design I (1DL300)

Date Thursday, October 24, 2013
Time 14:00 – 19:00
Teachers on duty:

- Matteo Magnani

Instructions:

Read through the complete exam and note any unclear directives before you start solving the questions.

Make sure you observe the following guidelines:

- You are allowed to use dictionaries to and from English and a calculator, but **no other material**.
- Assumptions outside of what is stated in the question must be explained. Any assumptions made should not alter the given question.
- NOTE! This examination contains **110** points in total and their distribution between questions is clearly identifiable. Note that you will get **credit only for answers that are correct**. To pass, you must score at least **60**. The examiner reserves the right to lower this score limit.
- Write legibly and clearly! Solutions failing to follow this guideline will be given zero points.
- Write your answer on only one side of the paper and use a new paper for each new question.
- Write your anonymous exam number on each page.
- Before handing in, sort all your answer pages by the order of the questions.

1) Questions on theory (12 points)

Indicate the correct answer. Every correct answer gives 3 points, every wrong answer gives -1. If you do not answer you do not get any points and you do not lose any. (The negative points have the effect that random answers will bring you an expected score of 0).

(a) With *physical data independence* we mean that:
   1) The physical organization of the data may change without affecting their logical representation, e.g., relation names and attributes remain the same.
   2) That data can be stored on storage devices that are independent of the client used to access the database management system, e.g., a disk on a remote server instead of the hard disk of your own laptop/desktop computer.
   3) That the way in which the data is saved in storage devices does not depend on the physical laws used by the specific device, e.g., magnetism (for disks and tapes), optics (for CDs and DVDs), and electrostatics (for main memories).
   4) None of the previous answers.

(b) If a table T has 10 rows, the SQL instruction: delete from T:
   1) Deletes the 10 rows, but does not remove the table from the database schema.
   2) Removes the table from the database schema (and as a consequence also the 10 rows).
   3) May delete less than 10 rows because of referential integrity constraints.
   4) None of the previous answers.

(c) In the relational model a candidate key is:
   1) An attribute (or set of attributes) where we do not allow NULL values.
   2) A minimal superkey.
   3) Any set of attributes whose values uniquely identify a tuple.
   4) None of the previous answers.

(d) In relational algebra the result of a difference between two relations r1 and r2 (assuming that the difference can be computed) contains a number of tuples |r1 – r2| that is:
   1) Always between 0 and |r1|.
   2) Always between 0 and |r2|.
   3) Always between min(|r1|,|r2|) and max(|r1|,|r2|).
   4) None of the previous answers.

   (notice that we use |r| to indicate the number of tuples in relation r and max(v1,v2) (resp., min) to indicate the higher (resp. lower) value among v1 and v2).

2) ER modeling (15 points)

A database for a higher education institution is being constructed with the following requirements.

1) We want to store the name, surname, telephone number and internal identifier of all the people (students and teachers).

2) A student cannot be registered as a teacher at the same time (if the same person is both teaching and studying s/he will be registered twice in the database, with two different internal identifiers).

3) Only for students, we want to know their age.

4) Every teacher belongs to a specific division of the institute, and receives a *division identification code* that is unique for that division (however, other divisions can use the same codes to identify their teachers).

5) Every division is identified by its name, and we also want to know its address.

Tasks:

Consider the following (incomplete) ER-diagram:

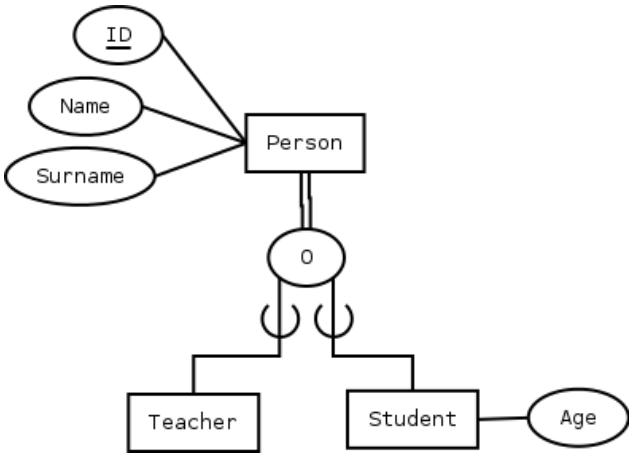

Figure 1

a) Can you identify any parts of the diagram not corresponding to the requirements (1, 2, 3) above, and correct them?

b) Complete the design of the ER diagram so that it satisfies all the requirements.

3) Translation to the Relational Model (12 points)

Consider again the ER diagram in Figure 1. (please notice that it might differ from the one that you developed in the previous exercise: for this exercise we want you to use exactly the diagram in Figure 1).

Tasks:

(a) Translate it to a relational model schema.
(b) Indicate all the primary keys.
(c) Indicate all the foreign keys, if any.
(d) Indicate all the attributes that allow NULL values, if any.

4) SQL (15 points)

Consider the following database schema:

Student(SID, Name, Surname, Age)

Registration(StudentID, CourseID)

Course(CID, Name, Cost)

Tasks:

(a) Write an SQL query to extract the identifiers of the people registered to at least one course costing more then 1000.

(b) Write an SQL query to extract the identifiers of the people spending more then 5000 in courses.

(c) Write an SQL query to extract the names of the courses with at least one registered student younger than 20 and at least one registered student older than 40. (Please notice that both conditions must be satisfied: a course with students younger than 20 but no student older than 40 must not appear in the result)

5) Relational Algebra (9 points)

Consider again the following database schema:

Student(SID, Name, Surname, Age)

Registration(StudentID, CourseID)

Course(CID, Name, Cost)

Tasks:

(a) Write a relational algebra expression to extract all the students registered to courses costing more than 1000. Return the name of the course and the name of the student.

(b) Write a relational algebra expression to extract the identifiers of all students registered to at least two courses. (Please notice that this query may not be easy – if you cannot solve it, do not get stuck here but focus on other exercises first then come back to this later)

6) Normalization (15 points)

A University is using the following *COURSE* table to store the information about courses and student grades.

COURSE	CNAME	YEAR	STUDENT	SNAME	GRADE
CS001	Database	2012	S001	Smith	A+
CS001	Database	2012	S002	Who	A
CS001	Database	2013	S003	Jensen	B
CS002	Algorithm	2013	S002	Who	A

Tasks:

a) List all the candidate keys.
b) List all the basic functional dependencies.
c) What Normal Form is this table in (1NF, 2NF, 3NF, BCNF)? Why? If some of FFDs violate the criteria for a certain normal form, identify them.
d) If the table is not in BCNF, present an example of a problem that can occur after updating a value in the table.
e) If the table is not in BCNF, present an example of a problem that can occur after removing one or more rows from the table.
f) If the table is not in BCNF, normalize this table such that it fulfills the requirements of BCNF. No extra attributes should be added. List the resulting tables and mark primary keys and foreign keys.
g) Populate the normalized tables with the data in the COURSE table [Hint: this task helps you to see whether or not your normalization is correct]

7) Physical Database Design (12 points)

Consider the following database schema: ten sensors are located at different places and record the temperature at that location. A new temperature is measured every second, generating a large number of records in the MEASURE relation.

```
SENSOR(SID, Brand, Location)
MEASURE(SID, Day, Time, Temperature)
```

Tasks:

a) Consider the SQL query:

```
SELECT Location, Temperature
FROM SENSOR NATURAL JOIN MEASURE
WHERE Day='2013-6-6';
```

Suggest an index that would speed up the execution of this query.
b) Write the SQL statement to create the suggested index.
c) Express two alternative query plans in relational algebra (used to indicate the order of operations performed to execute the SQL query).
d) Explain which plan would correspond to a faster execution, in your opinion, and when the index you suggested would be used by the system.

8) Transactions (12 points)

A bookstore uses a relational database to keep track of the available books. At some point they are left with a single copy of the book ”Luftslottet som sprängdes” (whose ID is B132). Therefore two new copies are ordered and inserted in the database by the bookstore administrator. However, one of the two new copies is damaged, so it is sold at a reduced price (10% discount).

```
INSERT INTO Book(ID,CopyID) VALUES('B132', 'C2');

SELECT Price INTO $price FROM Book WHERE ID='B132' AND CopyID='C1';

UPDATE Book SET Price=$price WHERE ID='B132' AND CopyID='C2';

INSERT INTO Book(ID,CopyID) VALUES('B132', 'C3');

UPDATE Book SET Price=$price*.9 WHERE ID='B132' AND CopyID='C3';
```

We call this transaction AD and summarize it as follows:

AD.WRITE(C2); AD.READ(C1); AD.WRITE(C2); AD.WRITE(C3); AD.WRITE(C3).

At the same time, an employee executes the following transaction EM to set a discount on all the current copies of two books:

```
UPDATE Book SET Price = Price*.9 WHERE ID='B133';

UPDATE Book SET Price = Price*.9 WHERE ID='B132';
```

We summarize this transaction as follows:

EM.WRITE(B133); EM.WRITE(B132).

Notice that the actual objects updated by this transaction depend on when it is executed. If the second update (the one indicated as EM.WRITE(B132)) is executed before AD, it corresponds to EM.WRITE(C1) because copies C2 and C3 have not inserted in the database yet. If it is executed after AD, it corresponds to EM.WRITE(C1); EM.WRITE(C2); EM.WRITE(C3).

Tasks:

a) Write the two serial schedules of these transactions and explain what their effect is.
b) Write a non-serial schedule that is serializable, and show why it is serializable.
c) Write a non-serializable schedule, and show why it is not serializable.
d) Explain why the schedule you proposed in the previous task (c) would not be allowed by a strict 2-phase locking protocol.

9) Database terminology (8 points)

a) What does “ACID” stand for in the concurrency control context?

b) What is the difference between a primary and a secondary index?