Advanced Course on Databases 2015
Exercises (10)

Submit your solutions on paper or by email before the examination that you plan to participate. The examination dates are: 23.11.2015, Jan 2016 and Feb 2016 (to be announced). At least 50% of the tasks should be solved acceptably.

Some of the tasks may require access to the textbook (Elmasri & Navathe: Fundamentals of Database Systems, Pearson/Addison-Wesley, 4th, 5th or 6th ed.)

Exercise 1

Explain the meanings of the following multi-level SQL-queries and transform them into single-level queries.

(a)

```
SELECT Pname, Plocation
FROM   PROJECT AS P
WHERE  EXISTS
 (SELECT *
  FROM   DEPARTMENT AS D
  WHERE  Dname = 'Administration'
  AND    P.Dnum = D.Dnumber)
```

(b)

```
SELECT Fname, Lname
FROM   EMPLOYEE AS E
WHERE  Dno IN
 (SELECT Dnumber
  FROM   DEPARTMENT
  WHERE  Mgr_ssn IN
   (SELECT Ssn
    FROM   EMPLOYEE AS M
    WHERE  E.Lname = M.Lname))
```

The queries are related to the COMPANY database, described in the course textbook (Figures 5.5-5.7 in the 4th and 5th ed., Figures 3.5-3.7 in the 6th edition). The relational schema is as follows:

EMPLOYEE (Fname, Minit, Lname, <u>Ssn</u>, Bdate, Address, Sex, Salary, Super_ssn, Dno)
DEPARTMENT (Dname, <u>Dnumber</u>, Mgr_ssn, Mgr_start_date)
DEPT_LOCATIONS (<u>Dnumber</u>, <u>Dlocation</u>)
PROJECT(Pname, <u>Pnumber</u>, Plocation, Dnum)
WORKS_ON (<u>Essn</u>, <u>Pno</u>, Hours)
DEPENDENT (<u>Essn</u>, <u>Dependent_name</u>, Sex, Bdate, Relationship)

Exercise 2

Assume that a relational database contains a relation of persons:

Person (<u>Pid</u>, Name, FatherPid, MotherPid)

where FatherPid and MotherPid are the identifiers of the father and mother of the current person. They are thus foreign keys, referring to the Person relation itself.

Write a query to determine all the descendants (children, grandchildren, etc.) of a person named "John Adams".
You may use either extended SQL or some free-form pseudo language for the task.

Exercise 3

Assume that we have two relations, R (<u>A</u>, B, C) and S (<u>C</u>, D), where R.A and S.C are primary keys, and R.C is a foreign key, referring to S.C.
The result of **natural join** of R and S is denoted by RS (<u>A</u>, B, C, D).
Assume that S has a primary B+-tree index on C, with depth = 3.
We also know the following quantities:

	R	S	RS
Number of tuples	1 000 000	100 000	1 000 000
Tuple size (bytes)	100	100	150

The disk block size is 4096 bytes, and the buffer size is 10 blocks.

Estimate the number of disk accesses for the following join algorithms.

(a) Sort-merge join
(b) Index-based join

(As for buffering, you can assume a worst-case situation, i.e. no probabilistic calculations are required.)

Exercise 4

Assume that we have the following relations in a university database:

```
Student(Sno, Sname, Address, MajorSubject)
Course (Cno, Cname, Credits)
Passed (Sno, Cno, Grade, Date)
```

with the obvious semantics. Assume further that the following SQL-query should be processed:

```
SELECT Credits, Grade
FROM   Student, Course, Passed
WHERE  Sname = 'Smith'
AND    Cname = 'Java'
AND    Student.Sno = Passed.Sno
AND    Course.Cno = Passed.Cno
```

Draw first the *canonical query tree* for this query and then convert the tree stepwise, by applying the *heuristic optimization* algorithm.
(For refinement of query trees, see the textbook chapter on Query Optimization)

Exercise 5

Below the numbers 1 to 4 refer to concurrent transactions, while x and y are some data units to be read (r) or written (w). Determine whether the following schedules are **conflict serializable** or not. For those which are, derive the corresponding serial schedule.

(a) $r_1(x); r_2(y); r_1(y); w_1(x); r_3(x); r_3(y); w_2(y); w_3(x); c_2; c_1; c_3;$

(b) $r_1(x); r_1(y); r_2(y); r_3(x); r_3(y); w_3(x); w_1(y); r_2(x); w_2(y); c_1; c_2; c_3;$

(c) $r_1(x); w_2(x); r_4(x); r_3(y); r_1(y); w_4(x); r_2(y); r_4(y); w_3(y); w_3(x); c_4; c_3; c_2; c_1;$

Exercise 6

Let us study the following schedule:

$r_3(Y)$; $r_3(Z)$; $r_1(X)$; $w_1(X)$; $w_3(Y)$; $w_3(Z)$; $r_2(Z)$; $r_1(Y)$; $w_1(Y)$; $r_2(Y)$; $w_2(Y)$; $r_2(X)$; $w_2(X)$;

Determine, whether the schedule is acceptable when using

(a) **2-phase locking protocol (2PL)**,
assuming that locks are reserved as late as possible, and released as early as possible. Assume further that upgrading and downgrading of locks is allowed.

(b) **Timestamp ordering protocol (TO)**,
assuming that the starting timestamps of transactions T_1, T_2, and T_3 are determined by their first operation in the schedule, and each operation takes one time unit. Thus, T_3 starts at time 0, T_1 at time 2, and so on.

Exercise 7

Describe the recovery process for the following log contents (schedule), assuming that the **immediate update protocol** (with checkpoints) is used. Specify, related to the system crash, which of the transactions (T_1, T_2, T_3, T_4) are rolled back, which operations are undone, and which are redone, and whether any cascading rollback takes place. The write_item entries in the log contain the before- and after-images (the numeric values) of the written data item (one of A to D).

[start_transaction, T_1]
[read_item, T_1, A]
[read_item, T_1, D]
[write_item, T_1, D, 10, 20]
[commit, T_1]
[checkpoint]
[start_transaction, T_2]
[read_item, T_2, B]
[write_item, T_2, B, 5, 12]
[start_transaction, T_4]
[read_item, T_4, D]
[write_item, T_4, D, 20, 15]
[start_transaction, T_3]
[write_item, T_3, C, 40, 30]
[read_item, T_4, A]
[write_item, T_4, A, 30, 20]
[commit, T_4]
[read_item, T_2, D]
[write_item, T_2, D, 15, 25]
--- System crash! ---

Exercise 8

[Textbook exercise] Suppose that we have the following multilevel relation in a database supporting mandatory multilevel access control. TC denotes the clearance level of the whole tuple.

EMPLOYEE

Name		Salary		JobPerformance		TC
Smith	U	40000	C	Fair	S	S
Smith	U	40000	C	Excellent	C	C
Jones	C	80000	S	Good	C	S

(a) How would this relation appear to a user with classification U?

(b) Suppose a user with classification U tries to update the salary of Smith to 50000. What would be the result of this action?

Exercise 9

Slide 200 of the lecture notes gives an example of a nested relation. On the basis of slides in Section 8, do the following:

(a) Present the same content using XML.

(b) Convert the XML representation into a normal relation using non-typed nodes.

Exercise 10

One of the recent NoSQL approaches to data modeling is a **column-oriented** database. Perform a web search and find answers to the following questions (no copy-paste, please):

(a) What is the basic idea and purpose of the column-oriented model.

(b) How does it differ from the normal relational model?

(c) In what kind of applications is it currently used?

(d) Mention some existing column-oriented database management systems.