The University of Birmingham
School of Computer Science

Solutions to Exercise Sheet 2

Question 1: Basic relational algebra

a. The tables generated in the calculation are:

$\pi_{X,Y}(T_1)$:

X	Y
x	15
x	10
y	15

$\pi_{X,D}(T_2)$:

X	D
x	p
x	t
y	p
y	r

$R = \pi_{X,Y}(T_1) \bowtie \pi_{X,D}(T_2)$:

X	Y	D
x	15	p
x	15	t
x	10	p
x	10	t
y	15	p
y	15	r

$\sigma_{D=p}(R)$

X	Y	D
x	15	p
x	10	p
y	15	p

Note that in the first table, there would have been another copy of the record x 15. But we have removed the duplicate copy since relational algebra insists that all tables should represent "relations", i.e., *sets* of records.

b. The selection operator $\sigma_{D=p}$ only depends on the table T_2. So, applying it early can reduce the size of the tables involved. Hence, a more efficient expresion is:

$$\pi_{X,Y}(T_1) \bowtie \pi_{X,D}(\sigma_{D=p}(T_2))$$

Question 2: Understanding relational algebra

a. The last name of all those members of staff whose first name is 'John'.

```
SELECT lastname
FROM staff
WHERE firstname='John';
```

b. The family name of all those members of staff who taught a course with more than 100 students (in any year).

```
SELECT DISTINCT lastname
FROM staff, lecturing
WHERE staff.sid = lecturing.sid AND numbers > 100;
```

c. The name of those courses, which had more than 100 students on them in at least one year, and which are not at level 1.

```
SELECT name
FROM lecturing, courses
WHERE lecturing.cid = courses.cid AND numbers>100
EXCEPT
SELECT name
FROM courses
WHERE level = 1;
```

d. List members of staff with the level 2 courses they taught in 1999.

```
SELECT lastname, name
FROM staff AS s, lecturing AS l, courses AS c
WHERE s.sid = l.sid AND c.cid = l.cid AND year = 1999 AND level = 2;
```

Question 3: From SQL to relational algebra

a. $\pi_{\texttt{name}}(\sigma_{(\texttt{year}=1999 \vee \texttt{year}=2000) \wedge \texttt{lastname}='\texttt{Jung}'}(\texttt{staff} \bowtie \texttt{lecturing} \bowtie \texttt{courses}))$

b. $\pi_{\texttt{name}}(\sigma_{\texttt{year}=2001}(\texttt{lecturing} \bowtie \texttt{courses})) - \pi_{\texttt{name}}(\sigma_{\texttt{year}=2000 \vee \texttt{year}=1999}(\texttt{lecturing} \bowtie \texttt{courses}))$

c. $\pi_{\texttt{name}}(\sigma_{\texttt{year}=1999 \wedge \texttt{year2}=2000}(\texttt{courses} \bowtie \texttt{lecturing} \bowtie \rho_{\texttt{year}\to\texttt{year2},\texttt{numbers}\to\texttt{numbers2}}(\texttt{lecturing})))$
(If you don't rename the `year` and `numbers` attributes, then you will just get "`lecturing`" again from the natural join "`lecturing` $\bowtie$ `lecturing`".)

Alternatively, you can project just the `cid` column from the `lecturing` tables so that there is no need for renaming: $\pi_{\texttt{name}}(\texttt{courses} \bowtie \pi_{\texttt{cid}}(\sigma_{\texttt{year}=1999}(\texttt{lecturing})) \bowtie \pi_{\texttt{cid}}(\sigma_{\texttt{year}=200}(\texttt{lecturing})))$

d. $\pi_{\texttt{lastname}}((\pi_{\texttt{sid}}(\texttt{staff}) - \pi_{\texttt{sid}}(\texttt{lecturing})) \bowtie \texttt{staff})$

Question 4: Suggesting functional dependencies

The pair (registration_number, date_of_inspection) is a candidate key. That means that we have the functional dependency

registration_number, date_of_inspection $\to X$

for every other attribute X. For instance, consider $X =$ garage.
Other possible dependencies:

owner_contact_phone	$\to$	owner
owner_contact_phone	$\to$	owner_address
registration_number	$\to$	model
registration_number	$\to$	year_of_first_registration
registration_number	$\to$	diesel_or_petrol
registration_number	$\to$	date_of_previous_MOT
model	$\to$	diesel_or_petrol
garage	$\to$	garage_address
garage_MOT_licence	$\to$	garage

Does the contact phone number determine the owner? Yes, as long as we assume that the owner does not give up his/her phone number, which then gets assigned to somebody else.
Car registration numbers (by which we mean the licence plate numbers) can be similarly transferred between vehicles, which might invalidate the stated dependencies.
When we say a model determines whether a car is diesel or petrol, we are assuming that the manufacturer has encoded that information in the model name, and we are using the full model name.
We are also assuming that the garage has not relocated to a new address during the time covered by the database.

Exercise 5: Outer Joins (Optional))

a. Inner join $T_1 \bowtie T_2$:

A	B	C	D
1	2	3	10
1	2	3	11

b. Dangling tuples of: T_1 :

A	B	C
4	5	6
7	8	9

T_2 :

B	C	D
2	6	10
6	7	12

c. Outer join $T_1 \overset{\circ}{\bowtie} T_2$:

A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	–
7	8	9	–
–	2	6	10
–	6	7	12

d. outer join of $T_1 \overset{\circ}{\bowtie}_{Left} T_2$:

A	B	C	D
1	2	3	10
1	2	3	11
4	5	6	–
7	8	9	–