Database Management Systems (COP 5725)

Spring 2014

Instructor:
Dr. Markus Schneider

TAs:
Soham Das
Yan Qiao
Yi Wang

Exam 2 Solutions

Name:	
UFID:	
Email Address:	

Pledge (Must be signed according to UF Honor Code)

On my honor, I have neither given nor received unauthorized aid in doing this assignment.

_____

Signature

For scoring use only:

	Maximum	Received
Question 1	20	
Question 2	25	
Question 3	30	
Question 4	25	
Total	100	

Question 1 (SQL and Relational Algebra) [20 points]

Let R(A:D1, B:D2, E:D3) and S(C:D1, H:D4) be two table schemas in which A, B, C, E, and H are attributes, A and C are the primary keys for the tables R and S respectively, D1, D2, D3, and D4 are data types, and F is a condition..

1. [6 points] Provide an equivalent (standard) Relational Algebra expression for the following SQL query.

```
SELECT  A
FROM    R
WHERE   A  IN  (SELECT  C
                FROM    S
                WHERE   F)
```

$\pi_A(\sigma_{A=C}(R \times \sigma_F(S)))$

2. [8 points] Provide an equivalent (standard) Relational Algebra expression for the following SQL query.

```
SELECT  A
FROM    R
WHERE   A > ALL (SELECT  C
                 FROM    S
                 WHERE   F)
```

$\pi_A(R) - \pi_A(\sigma_{A \leq C}(\pi_A(R) \times \pi_C(\sigma_F(S))))$

3. [6 points] Is the following SQL statement equivalent to the relational algebra expression? Please provide yes or no, and explain why.

SELECT B FROM R WHERE F

and

$$\pi_B(\sigma_F(R))$$

No. Because the SQL results may have duplications, whereas the relational algebra results are sets (no duplications).

Question 2 (QBE) [25 points]

Consider the following schema:

Movie(MID, Director, MName, Duration, Year)
Celebrity(CID, CName, YearOfBirth)
ActsIn(CID, MID, Salary)
Roles(CID, MID, RoleName)

Note: Director is a foreign key referencing Celebrity.CID.
Draw QBE tables for the following exercises.

1. [5 points] Find the movies that are directed by Ang Lee in which Suraj Sharma is acting.

Movie	MID	Director	Name	Duration	Year
	P._X	_Z			

ActIn	CID	MID	Salary
	_Y	_X	

Celebrity	CID	CName	YearOfBirth
	_Y	Suraj Sharma	
	_Z	Ang Lee	

2. [5 points] Find the actor or actress who has played at least two roles in one movie.

Roles	CID	MID	RoleName
	P._X	_Y	_Z
	_X	_Y	¬ _Z

3. [5 points] Find the actor or actress who has the most total salary.

ActIn	CID	MID	Salary
	P.G._X		SUM.ALL._Y
¬	G._Z		SUM.ALL._S

Condition
SUM.ALL._S > SUM.ALL._Y

4. [5 points] Delete the longest movie that is shot in 1990's

Movie	MID	Director	Name	Duration	Year
D.				_X	_Y
¬				>_X	_Z

Condition
_Y <2000 AND _Y>1989 AND _ Z <2000 AND _Z>1989

5. [5 points] List the names of movies that Will Smith has played in grouped by year.

Movie	MID	Director	Name	Duration	Year
	_Y		P.		G.

ActIn	CID	MID	Salary
	_X	_Y	

Celebrity	CID	CName	YearOfBirth
	_X	Will Smith	

Question 3 (SQL) [30 points]

Consider the following database schema:

Product (maker, model, type)
Desktop (model, speed, ram, hd, rd, price)
Laptop (model, speed, ram, hd, screen, price)
Printer (model, color, type, price)

The Product relation gives the manufacturer, model number and type (desktop, laptop, or printer) of various products. We assume for convenience that model numbers are unique over all manufacturers and product types. The Desktop relation gives for each model number that is a Desktop its speed (of the processor, in gigahertz), the amount of RAM (in megabytes), the size of the hard disk (in gigabytes), the type of the removable disk (CD, DVD or Blue-ray), and its price. The Laptop relation is similar, except that the screen size (in inches) is recorded in place of the removable disk. The Printer relation records for each printer model whether the printer produces color output (true if so), the process type (laser, ink-jet, or bubble), and the price.

Write a SQL statement for the following questions:

(1) [5 points] Find the manufacturers that make at least 10 different models of Desktops or Laptops combined.

Solution:

```
SELECT maker
FROM Product
WHERE type = 'laptop' OR type = 'desktop'
GROUP BY maker
HAVING COUNT (model) >= 10;
```

(2) [5 points] Find the average hard disk size of a desktop for all those manufacturers that make printers.

Solution:

```
SELECT AVG (D.hd) AS Avg_HD_Size
FROM Product R, Desktop D
WHERE R.model = D.model AND R.maker IN (
    SELECT maker
    FROM Product
    WHERE type = 'printer'
);
```

(3) [5 points] Find the model number and price of the computer (desktop or laptop) with the highest price.

Solution:

```
SELECT model, price FROM (
    SELECT model, price FROM Desktop UNION
    SELECT model, price FROM Laptop
    ) M1
WHERE M1.price >= ALL (
    SELECT price FROM Desktop UNION
    SELECT price FROM Laptop
);
```

(4) [5 points] Delete all laptops in the Laptop relation made by a manufacturer that doesn't make any printers.

Solution:

```
DELETE FROM Laptop L
WHERE NOT EXISTS (
    SELECT * FROM Product P1, Product P2
    WHERE P1.model = L.model AND P1.maker = P2.maker
        AND P2.type = 'printer'
);
```

(5) [10 points] Reduce the price of a printer by 10% if it is made by a manufacturer that makes the cheapest desktop or the cheapest laptop.

Solution:

```
UPDATE Printer
SET price = price*0.9
WHERE model IN (
    SELECT P1.model
    FROM Product P1
    WHERE P1.maker IN (
        SELECT D.maker
        FROM Desktop D
        WHERE D.price = (SELECT MIN (price) FROM Desktop)
        UNION
        SELECT L.maker
        FROM Laptop L
        WHERE L.price = (SELECT MIN (price) FROM Laptop)
    )
);
```

Question 4 (Relational Algebra and SQL) [25 points]

Consider a simplified Twitter like system with the following specifications:

- Users post 'tweets', which are short pieces of text
- They may tag their tweets with zero or more tags of their own choice. For example, a user tweeting about the Gators may decide to use the tag 'Gators'.
- A user may follow zero or more other users, which means that their 'tweets' are visible to this user when he/she logs in.

Consider the following database schema (primary keys are underlined):

- Person (pname, city, ttext) – Assuming pname is unique
- Follows (pname1, pname2) – Person pname1 follows person pname2
- Tweets (tid, ttitle, ttext) – Tweet with tid has title ttitle and text ttext
- PersonTweets (pname, tid, ts) – Person pname posted tweet tid at timestamp ts
- TweetTag (tid, tagname) – Tweet tid had tagname in its list of tags

For your convenience, you can use the follow abbreviations:
P for Person, F for Follows, T for Tweets, PT for PersionTweets, TT for TweetTag.

Express the following queries in relational algebra and in SQL.

(1) [5 points] Find all the tags 'Bill Gates' used in his tweets. (i.e. tweeting interests)

Solution: $\pi_{tagname}(\sigma_{pname='Bill\ Gates' \wedge PT.tid=TT.tid}(PT \bowtie TT))$

```
SELECT tagname
FROM TT, PT
WHERE PT.tid = TT.tid AND pname = 'Bill Gates';
```

(2) [5 points] Find all the tags 'Bill Gates' reads in the tweets of the people he follows. (i.e. reading interests)

Solution: $\pi_{tagname}(\sigma_{pname='Bill\ Gates' \wedge PT.tid=TT.tid \wedge F.pname2 = PT.pname}(F \bowtie PT \bowtie TT))$

```
SELECT tagname
FROM TT, PT, F
WHERE F.pname2 = PT.pname AND PT.pname = 'Bill Gates' AND PT.tid = TT.tid;
```

(3) [5 points] Find all the people (pname and city) who follow people who follow 'Bill Gates'. (i.e. second-level followers)

Solution: $Followers = \pi_{pname1}(\sigma_{pname2= \text{'Bill Gates'}}(F))$

$\pi_{pname,city}(\sigma_{pname1=pname}(P \bowtie (\sigma_{pname1=pname2}(Followers \bowtie F))$

```
SELECT pname, city
FROM F, P
WHERE P.pname = F.pname1 AND F.pname2 IN (
        SELECT F2.pname1
        FROM F F2
        WHERE F2.pname2 = 'Bill Gates'
        );
```

(4) [5 points] Find all the people (pname and city) who follow at least everyone that 'Bill Gates' follows.

Solution: $\pi_{pname,city}(\sigma_{pname1=pname}(P \bowtie (F \div \pi_{pname2}(\sigma_{pname1= \text{'Bill Gates'}}(F))))$

```
SELECT pname, city FROM P
WHERE NOT EXIST (
        (SELECT F1.pname2 FROM F F1
        WHERE F1.pname1 = 'Bill Gates')
        EXCEPT
        (SELECT F2.pname2 FROM F F2
        WHERE F2.pname1 = P.pname)
);
```

(5) [5 points] Find all pairs of people (pname) who have at least one follower in common. (assume <, =, and > can be used for string comparison).

Solution:

$\pi_{F.pname2,F1.pname2}(\sigma_{F.pname1=F1.pname1 \wedge F.pname2<F1.pname2}(F \bowtie \rho_{F1}(F)))$

```
SELECT F1.pname2, F2.pname2
FROM F F1, F F2
WHERE F1.pname1 = F2.pname1 AND F1.pname2 < F2.pname2;
```