Assignment 4

Due Wednesday, October 10 2018 by 11:59pm

In this assignment, you will be required to use PostgreSQL. Your solutions should include the PostgreSQL statements for solving the problems. You will use the same data as that used in Assignment 3.

Remark: Consider a relation $R(A, B)$ and a relation $S(C)$ and consider the following RA (Relational Algebra) expression F:

$$\pi_A(R) - \pi_A(\sigma_{B=1}(R \bowtie_{B=C} S))$$

Then we can write this query in SQL in a variety of ways that closely mimic its RA formulation. One way to write this RA expression in SQL is as follows:

```
SELECT  A
FROM    R
EXCEPT
SELECT A
FROM (SELECT R.A, R.B, S.C
      FROM  R INNER JOIN S ON B=C
      WHERE  B = 1) q;
```

An alternative way is to use the WITH statement of SQL.[1] First, we separate the RA expression F into sub-expressions as follows. (In this case, notice that each sub-expression corresponds to the application of a single RA operation. More generally, one can of course use sub-expressions that can contain multiple RA operations.)

Expression Name	RA expression
E_1	$\pi_A(R)$
E_2	$R \bowtie_{B=C} S$
E_3	$\sigma_{B=1}(E_2)$
E_4	$\pi_A(E_3)$
F	$E_1 - E_4$

---

[1] This is especially so if the RA expression is long and complicated.

Then we write the following SQL query. Notice how the expressions $E1$, $E2$, $E3$, and $E4$ occur as separate queries in the WITH statement and that the final query gives the result for the expression F.[2]

If you use overloading of the relational name, the query becomes

```
WITH
E1 AS (SELECT DISTINCT A FROM R),
E2 AS (SELECT A, B, C FROM (R INNER JOIN S ON (B = C)) e2),
E3 AS (SELECT A, B, C FROM E2 WHERE B = 1),
E4 AS (SELECT DISTINCT A FROM E3)
(SELECT A FROM E1) EXCEPT (SELECT A FROM E4);
```

In your answer to a problem, you may write the resulting RA expression with or without the WITH statement. (Your SQL query should of course closely resemble the RA expression it is aimed to express.) In a separate file you should also submit the text for the RA expressions in their standard notation, just as illustrated for the expression F above.

1. (10 points) Let $W(A, B)$ be a relation schema. The domain of A is INTEGER and the domain of B is VARCHAR(5).

   Write a RA expression which returns the A-values of tuples in W if A is a primary key of W. Otherwise, i.e., if A is not a primary key, then your query should return the A-values of tuples in W for which the primary key property is violated.

   For example if W is as follows

   $$W$$

A	B
1	John
2	Ellen
3	Ann

   then your RA expression should return the following answer since, in this case, A satisfies the primary property for W.

   ```
   a
   ---
   1
   2
   3
   (3 rows)
   ```

---

[2]For better readability, I have used relational-name overloading. Sometimes, you may need to introduce new attribute names in SELECT clauses using the AS clause. Also, use DISTINCT were needed.

However, if we W is as follows

$$W$$

A	B
1	John
2	Ellen
2	Linda
3	Ann
4	Ann
4	Nick
4	Vince
4	Lisa

then your RA expression should return the following answer because the primary key property of A for W is violated for the A-values 2 and 4.

```
a
---
2
4
(2 rows)
```

2. In the following questions, use the files student.txt, majors.txt, book.txt, cites.txt, and buys.txt that are provided for this assignment. Use the same relations as in Assignment 3.

   Write the following queries as RA expressions in the standard RA notation. Submit these queries as a separate document.

   Then, for each such RA expression, write a SQL query (possibly using the WITH statement) that mimics this expression.

   (a) (10 points) Find the sid and sname of each student who bought a book that cites another book.

   (b) (10 points) Find the sid and sname of each student who has at least two majors.

   (c) (10 points) Find the sid of each student who bought exactly one book.

   (d) (10 points) Find the bookno and title of each book with the second to lowest price.

   (e) (10 points) Find the bookno and title of each book that was only bought by the student with sid = 1001.

   (f) (10 points) Find the sid and sname of each student who bought at least two books that cost less than $50.

3

(g) (10 points) Find the bookno of each book that was not bought by all students who major in CS.

(h) (10 points) Find the bookno of each book that is not cited by a book that cost more than $50.

(i) (10 points) Find the sid of each student who not only bought books that cost less than $30.

(j) (10 point) Find each pair (s, b) such that s is the sid of a student who bought a book that does not cite the book with bookno b.

(k) (10 points) Find the pair of different booknos (b_1, b_2) that where bought by the same CS students.

(l) (10 points) Find the pairs of different sid (s1,s2) of students such that all books bought by student s1 were also bought by student s2.

(m) (10 points) Find the bookno of each book that is cited by all but one book.

3. Translate each of the following SQL queries in RA expressions using the translation algorithm given in class. Make sure that these RA expressions are formulated in SQL with RA operations so they can be run in PostgreSQL. You are required to show the intermediate steps that you took during the translation.

In addition, submit separately the RA expressions in the standard RA notation.

(a) Find the sid and major of each student who bought a book that cost less than $20.

```
select m.sid, m.major
from   major m
where  m.sid in (select t.sid
                 from   buys t, book b
                 where  t.bookno = b.bookno and b.price < 20);
```

(b) Find each (s, b) pair where s is the sid of a student and where b is the bookno of a book whose price is the cheapest among the books bought by that student.

```
select distinct t.sid, b.bookno
from   buys t, book b
where  t.bookno = b.bookno and
       b.price <= ALL (select b1.price
                       from   buys t1, book b1
                       where t1.bookno = b1.bookno and t1.sid = t.sid);
```

4

(c) Find the bookno and title of each book that cost between \$20 and \$40 and that is cited by another book.

```
select b.bookno, b.title
from   book b
where  20 <= b.price and b.price <= 40 and
       b.bookno in (select c.citedbookno
                    from   cites c);
```

(d) Find the sid and name of each student who majors in 'CS' and who bought a book that is cited by a lower priced book.

```
select s.sid, s.sname
from   student s
where  s.sid in (select m.sid from major m where m.major = 'CS') AND
       exists (select 1
               from   buys t, cites c, book b1, book b2
               where  s.sid = t.sid and t.bookno = c.citedbookno and
                      c.citedbookno = b1.bookno and c.bookno = b2.bookno and
                      b1.price > b2.price);
```

(e) Find the bookno and title of each book that is not bought by all students who major in 'CS'.

```
select b.bookno, b.title
from   book b
where  exists (select m.sid
               from   major m
               where  m.major = 'CS' and
                      m.sid not in(select t.sid
                                   from   buys t
                                   where  t.bookno = b.bookno));
```

(f) Find the bookno and title of each book that is bought by all students who major in both 'CS' and in 'Math'.

```
select b.bookno, b.title
from   book b
where  not exists (select s.sid
                   from   student s
                   where  s.sid in (select m.sid from major m
                                    where m.major = 'CS') and
                          s.sid in (select m.sid from major m
                                    where m.major = 'Math') and
                          s.sid not in (select t.sid
                                        from   buys t
                                        where  t.bookno = b.bookno));
```