# CMPT 354 Spring 2018
# Database Systems
# Martin Ester
# TA: Hongwei Liang

## Midterm Exam with Solution

Total marks: 100 (10% of the class)

**Problem 1: RA queries (40 marks)**

Consider the following schema of a University database:

```
Student(sid: integer, name: string, address: string, gpa: real)
Enrollment(sid: integer, cid: integer, semester: integer,
           grade: real)
Section(cid: integer, semester: integer, taught_by: integer)
Course(cid: integer, title: string, department: string)
```

The relation Enrollment records that a Student sid enrolled in Section (cid, semester) and obtained the given grade. The primary key of a relation consists of all its underlined attributes. There are foreign key constraints between (Enrollment.cid, Enrollment.semester) and (Section.cid, Section.semester), Enrollment.sid and Student.sid, and Section.cid and Course.cid.

Formulate the following queries in Relational Algebra.

a) Find the names of Students who have taken at least one Section with a grade of at least 2.0.

$$\pi_{sname}((\sigma_{grade \geq 2.0} Enrollment) \infty Student)$$

b) Find the names of Students who have taken all Courses offered by department "CMPT".

$$\pi_{name}((\pi_{sid,cid}(Enrollment) / \pi_{cid}(\sigma_{department="CMPT"} Course)) \infty Student)$$

c) Find the cid of Courses that have been taught in some semester by Instructor 1 and in some semester by Instructor 2 but never by Instructor 3.

$$\rho_{Cids1}(\pi_{cid}(\sigma_{taught\_by=1} Section))$$
$$\rho_{Cids2}(\pi_{cid}(\sigma_{taught\_by=2} Section))$$
$$\rho_{Cids3}(\pi_{cid}(\sigma_{taught\_by=3} Section))$$
$$(Cids1 \cap Cids2) - Cids3$$

d) Find the Students who have taken at least three different Courses (with different cid).

$$\rho_{AllTriples(1->sid1,2->cid1,5->sid2,6->cid2,9->sid3,10->cid3)}(Enrollment \times Enrollment \times Enrollment)$$
$$\rho_{SelectedTriples}(\sigma_{sid1=sid2\ AND\ sid1=sid3\ AND\ cid1 \neq cid2\ AND\ cid1 \neq cid3\ AND\ cid2 \neq cid3} AllTriples)$$
$$\pi_{name}(SelectedTriples \infty_{sid1=sid} Student)$$

**Problem 2: SQL queries (40 marks)**

Consider the University database from Problem 1.

Formulate the folliwing queries in SQL. You can use either Standard SQL or SQL Server SQL. Write SQL keywords in CAPITAL letters.

a) Find the names of Students who have taken at least one Section with a grade of at least 2.0.

```
SELECT S.name
FROM Student S, Enrollment E
WHERE S.sid = E.sid AND E.grade>=2.0;
```

b) Find the names of Students who have taken all Courses taught by Instructor 7.

```
SELECT S.name
FROM Student S
WHERE NOT EXISTS
        ((SELECT S.cid
         FROM Section S
         WHERE S.taught_by = 7)
        EXCEPT
         (SELECT E.cid
          FROM Enrollment E
          WHERE E.sid = S.sid AND E.cid = S.cid));
```

Alternative (Students who have taken all Sections taught by Instructor 7):

```
SELECT S.name
FROM Student S
WHERE NOT EXISTS
        (SELECT *
        FROM Section S
        WHERE S.taught_by = 7 AND NOT EXISTS
         (SELECT *
          FROM Enrollment E
          WHERE E.sid = S.sid AND E.cid = S.cid));
```

c) Find the cid of Courses that have been taught in some semester by Instructor 1 and in some semester by Instructor 2 but never by Instructor 3.

```
((SELECT C.cid
FROM Course C
WHERE C.taught_by = 1)
INTERSECT
(SELECT C.cid
FROM Course C
WHERE C.taught_by = 2))
EXECPT
(SELECT C.cid
FROM Course C
WHERE C.taught_by = 3);
```

d) Find the sid and the number of the Courses for Students who have taken at least three different Courses (with different cid) with a grade of at least 2.0.

```
SELECT E.sid, COUNT(DISTINCT cid)
FROM Enrollment E
WHERE E.grade>=2.0
GROUP BY S.sid
HAVING COUNT(DISTINCT cid) >= 3;
```

**Problem 3: Multiple Choice Questions (20 marks)**

Mark your answers for the following questions with a BIG "X" in the row directly under the question. Note that multiple answers may be correct. Here are the rules for marking:

(1) For every question, mark at least one answer. If you do not mark any answer, you get 0 marks for that question.
(2) If you mark a wrong answer, you get 0 marks.
(3) If you mark at least one correct answer (and no wrong answer) but miss some correct answer(s), you get 1 mark.
(4) You get 2 marks if you mark all correct answers (but no wrong answer).

| | | | | |
|---|---|---|---|---|
| a) Given a relation R (X). A key is a functional dependency K -> Y where . . . | K = X | **K ⊆ X** | **Y = X** | Y ⊂ X |
| Answer(s) to a) | | | | |
| b) Given the functional dependencies X->Y and Y->Z, we know that . . . | **X->Z holds** | X->Z does not hold | Y->X holds | **Y->X may hold** |
| Answer(s) to b) | | | | |
| c) If a relation R(A,B,C) with the only key A is in BCNF, then the functional dependency. . | **A->B is possible** | A->B is impossible | B->C is possible | **B->C is impossible** |
| Answer(s) to c) | | | | |
| d) Every non-empty Entity Relationship diagram . . . | Contains at least one relationship set | **Contains at least one entity set** | Contains at least two attributes | Contains at least three attributes |
| Answer(s) to d) | | | | |
| e) When translating an Entity | Every relationship set | Every entity set is translated | **Most entity sets are translated** | Every relationship set |

sub class

| | | | | |
|---|---|---|---|---|
| Relationship diagram into a relational schema . . . | is translated into two tables | into a table | **into a table** | is translated into a table |
| Answer(s) to e) | | | | |
| f) In ER design, when choosing between an attribute and an entity set to model an address, . . . | Always prefer an attribute | **Prefer an entity set if the address can have multiple values** | Prefer an attribute if the address values are very short | **Prefer an entity set if need to access components of the address** |
| Answer(s) to f) | | | | |
| g) We want to avoid redundancies . . . | **to minimize storage costs** | to optimize query processing | **to avoid update anomalies** | to maximize the number of queries performed concurrently |
| Answer(s) to g) | | | | |
| h) The ACID principle requires that . . . | All of the actions of a transaction are performed | A trasaction transforms an inconsistent database state into a consistent one | Onle one transaction is executed at a time | **Either all or none of the actions of a transaction are performed** |
| Answer(s) to h) | | | | |
| i) The following SQL statement modifies the state of a database instance . . . | **INSERT** | **DELETE** | SELECT . . . FROM . . . WHERE . . . | CREATE TABLE |
| Answer(s) to i) | | | | |
| k) People use a database system because . . . | queries are always more efficient than when executed directly on the file system | it can be programmed at a low level | **it provides crash recovery** | it requires less storage space than a file system-based implementation |
| Answer(s) to k) | | | | |

Need space to store description