

NANYANG TECHNOLOGICAL UNIVERSITY

SEMESTER 2 EXAMINATION 2015-2016

CZ2007 – INTRODUCTION TO DATABASES

Apr/May 2016

Time Allowed: 2 hours

INSTRUCTIONS

1. This paper contains 4 questions and comprises 7 pages.
 2. Answer **ALL** questions.
 3. This is a closed-book examination.
 4. All questions carry equal marks.
-

1. (a) Consider a database for a telephone service provider, TelSing, with the following requirements:
 - Each customer is uniquely identified by an ID, and has a name, a gender, a date of birth, and an address.
 - Each customer has one or more phone numbers. Each phone number is associated with a phone plan, which has a unique ID and is either prepaid or postpaid. A postpaid plan has three quotas per month, for local calls (in minutes), international calls (in minutes), and internet data usage (in KBs), respectively. A prepaid plan has three charge rates, for local calls (per minute), international calls (per minute), and internet data usage (per KB), respectively.
 - For each phone number, TelSing records the start and end dates of its current phone plan. For each phone number on a postpaid plan, TelSing records its remaining quotas in the current month in terms of local calls, international calls, and data usage. For each phone number on a prepaid plan, TelSing records its current account balance (in Singapore dollars).

Question 1(a) continues on Page 2

- At any given point in time, a phone number is registered under at most one customer, but it may be registered under different customers in different periods of time. For each phone number, TelSing keeps a list of the customers that have ever been associated with the number. For each association, TelSing records the start and end dates.

- (i) Construct an ER diagram that captures the requirements as much as possible.

(10 marks)

- (ii) Convert the ER diagram into a set of tables. Indicate the primary keys.

(4 marks)

- (b) Consider a movie database with the following schema:

ACTOR(AID, AName, Gender, DOB)
MOVIE(MID, MName, Year, Profit)
ROLE(AID, MID, RoleName, Pay)

ACTOR and MOVIE record information about actors and movies, respectively. Whenever an actor is cast in a movie, the ROLE table records the actor's role and pay in the movie. ROLE.AID and ROLE.MID are foreign keys referencing ACTOR.AID and MOVIE.MID, respectively.

Answer each of the following queries with relational algebra. You may use the following operators:

σ (selection), Π (projection), \cup (union), \cap (intersection), $-$ (intersection), γ (grouping and aggregation), δ (duplicate elimination), \div (division), $:=$ (assignment), ρ (rename)

- (i) Find the names of the female actors who have appeared in at least one movie with a profit over 1,000,000.

(2 marks)

- (ii) For every actor, list the total number of movies that he/she appeared in from 2005 to 2015, as well as the total pay that he/she received during this period.

(2 marks)

Question 1(b) continues on Page 3

- (iii) Find the actors who have only appeared in movies with negative profits (i.e., the profits of those movies are smaller than zero).

(3 marks)

- (iv) Find the actors who have received at least 1,000,000 total pay in 2015, and have appeared in at most three movies with negative profits (over all years).

(4 marks)

2. Consider a relation $R(A, B, C, D, E)$ with the following functional dependencies: $AB \rightarrow C$, $AD \rightarrow B$, $BD \rightarrow E$, $DE \rightarrow B$, $E \rightarrow D$.

- (a) Verify whether R is in BCNF. If R is not in BCNF, apply an BCNF decomposition on R , and then verify whether your BCNF decomposition preserves all functional dependencies.

(13 marks)

- (b) Verify whether R is in 3NF. If R is not in 3NF, apply a 3NF decomposition on R .

(12 marks)

3. (a) Consider the following relational schema.

```
PATIENTS(patientId, gender, birthdate)
MEDITEMS(itemId, label)
MEDEVENTS(patientId, itemId)
```

Consider the following two queries. Do they always produce same results? Justify your answer.

Query 1:

```
SELECT      DISTINCT patientId
FROM        PATIENTS
WHERE       PATIENTS.patientId NOT IN
            (SELECT  patientId
             FROM      MEDEVENTS AS me
                   JOIN MEDITEMS AS mi
                   ON   me.itemId=mi.itemId
             WHERE     mi.label LIKE '%digoxin%')
```

Question 3(a) continues on Page 4

Query 2:

```
SELECT    DISTINCT patientId
FROM      PATIENTS
WHERE     PATIENTS.patientId IN
          (SELECT  patientId
           FROM     MEDEVENTS JOIN MEDITEMS
           ON MEDEVENTS.itemId =
            MEDITEMS.itemId AND MEDITEMS.label
            NOT LIKE '%digoxin%')
```

(5 marks)

- (b) Consider the following relational schema.

```
ACCOUNT(accountNumber, branchName, balance)
BRANCH(branchName, street, city, assets)
CUSTOMER(customerId, street, city)
DEPOSIT(customerId, accountNumber, balance)
```

- (i) Define a **view** KeyBranch that gives for each branch its branchName, city, and assets. The branch should have more than 50 accounts and the total balance of all accounts is greater than \$1,000,000.

(5 marks)

- (ii) Suppose we want to check that, for each branch, the total balance of all accounts is less than or equal to the assets of the branch. Write an **assertion** statement to enforce this constraint.

(5 marks)

- (c) Consider the simple relation Employee(ID, salary) storing the employee Ids and salaries, where ID is a key. Consider the following two triggers over this relation:

Question 3(c) continues on Page 5

Trigger T1:

```
CREATE TRIGGER T1
AFTER INSERT ON Employee
REFERENCING NEW as New_Emp
FOR EACH ROW
UPDATE Employee
SET salary = 1.1 * (SELECT max(salary)
                    FROM Employee)
WHERE ID = New_Emp.ID
```

Trigger T2:

```
CREATE TRIGGER T2
AFTER INSERT ON Employee
REFERENCING NEWTABLE AS New_Emp
FOR EACH STATEMENT
UPDATE Employee
SET salary = 1.1 * (SELECT max(salary)
                    FROM Employee)
WHERE ID IN (SELECT ID FROM New_Emp)
```

Assume that relation Employee has no tuple initially. Suppose that we had inserted the following four rows into the Employee table as the result of a single SQL statement:

```
1 1000
2 2000
3 3000
4 4000
```

- (i) Show the final database state after trigger execution if only trigger T1 is defined.
(3 marks)
- (ii) Show the final database state after trigger execution if only trigger T2 is defined.
(3 marks)

- (d) Consider the following fragment of a JDBC program. Is there anything wrong with this program? Justify your answer.

```
String code;
/* Get value for code from user */

String myQuery = "SELECT name" + "FROM Courses"
+ "WHERE code = " + code;

Statement myStmt = myCon.createStatement();
ResultSet rs = myStmt.executeQuery(myQuery);
/* Do something with result. */
```

(4 marks)

4. (a) Consider the following database schema:

```
EMP(SSN, name birthdate, street, city, DNo,
SuperSSN, Salary)
DEPT(Dnum, Dname, MgrSSN)
```

In the above relations, SuperSSN, MgrSSN are foreign keys pointing to SSN, and DNo is a foreign key pointing to Dnum.

Give an SQL expression for the following query. Your solution should be only one SQL statement.

Find the department number and name of the department with the largest average salary (of its employees).

(6 marks)

- (b) Consider the XML document **result.xml** in Figure Q4b.

- (i) Write a Document Type Definition (DTD) for the XML document in Figure Q4b.

(6 marks)

- (ii) The flexibility of XML enables us to nest elements in a more natural way than in the document shown in Figure Q4b. Write an XML document that contains the same information as in **result.xml**, but which uses nesting, and avoids duplication of applicant identifiers (i.e., appNum values).

(8 marks)

Question 4(b) continues on Page 7

```
<results>
  <applicants>
    <applicant name="Doreen" appNum="a1"/>
    <applicant name="Dilwyn" appNum="a2"/>
    <applicant name="Suzanne" appNum="a3"/>
  </applicants>
  <choices>
    <choice applicant="a1" code="MPSOF"
    choiceNum="1" meritScore="750"/>
    <choice applicant="a1" code="MPALG"
    choiceNum="2" meritScore="750"/>
    <choice applicant="a1" code="MPCSN"
    choiceNum="3" meritScore="800"/>
    <choice applicant="a2" code="MPALG"
    choiceNum="1" meritScore="700"/>
    <choice applicant="a3" code="MPCSN"
    choiceNum="1" meritScore="850"/>
    <choice applicant="a3" code="MPALG"
    choiceNum="2" meritScore="850"/>
  </choices>
</results>
```

Figure Q4b

- (c) Consider the **result.xml** in Figure Q4b. Write XPath expressions for the following queries posed on this document.
- (i) Find all choice elements where the choice number (choiceNum) is 1 and the merit score is greater than 800.
(2 marks)
 - (ii) Find all code of applicant "Doreen".
(3 marks)

END OF PAPER

CZ2007 INTRODUCTION TO DATABASES

Please read the following instructions carefully:

- 1. Please do not turn over the question paper until you are told to do so. Disciplinary action may be taken against you if you do so.**
2. You are not allowed to leave the examination hall unless accompanied by an invigilator. You may raise your hand if you need to communicate with the invigilator.
3. Please write your Matriculation Number on the front of the answer book.
4. Please indicate clearly in the answer book (at the appropriate place) if you are continuing the answer to a question elsewhere in the book.