

UNIVERSITI SAINS MALAYSIA

First Semester Examination  
Academic Session 1997/98

September 1997

**CIS401 - Database Management and Design**

Duration : [3 hours]

---

**INSTRUCTION TO CANDIDATE:**

- Please ensure that this examination paper contains **EIGHT** questions in **FIVE** printed pages before you start the examination.
  - Answer **ALL** questions. If you choose to answer the questions in English, at least one question must be answered in Bahasa Malaysia.
- 

ENGLISH VERSION OF THE QUESTION PAPER

1. (a) What is the purpose of authentication? Describe the authentication schemes you know. (50/100)

- (b) For each of the following, write SQL statements to create views where needed and to grant the indicated privileges for the following database:

Person (PERSID, PERSNAME, ADDRESS)  
 Car (LICENSE, YEAR, MODEL)  
 Accident (DATE, DRIVER, KIND-OF-ACCIDENT, DAMAGE-AMOUNT)  
 Owns (PERSID, LICENSE)

- (i) Give permission to read the tables *Person* and *Car* to user U1. The user may also read *Accident*, but without the DAMAGE-AMOUNT attribute.
- (ii) Give permission to read and modify the tables *Car* and *Accident* to user U2. This user may authorise user U3 to read and modify *Accident*, but not *Car*.
- (iii) Give permission to read and modify the tables *Person*, *Accident*, and *Car* to user U4. This user may authorise users U5 to read these tables and to grant this right to other users. (50/100)

2. (a) Describe what is a trigger and what does it serve for, give an example. (40/100)

- (b) Define the following database with the use of domain constraints and entity integrity constraints:

STUDENT (STUDID, STUDNAME, MAJOR, CREDITS)  
 FACULTY (FACID, FSCNAME, DEPT, RANK)  
 CLASS (COURSE#, FACID, TIME, ROOM)  
 ENROLL (COURSE#, STUDID, GRADE)

(30/100)

- (c) Write assertions stating that:

- (i) for each pair of COURSE# and STUDID values in relation ENROLL, there is a corresponding STUDID value in relation STUDENT and COURSE# value in ENROLL.
- (ii) there are no two classes assigned to the same room at the same time. (30/100)

3. Consider the following two transactions:

```
T1: read(A);
    read(B);
    if A = 0 then B := B+1;
    write(B);
T2: read(B);
    read(A);
    if B = 0 then A := A+1;
    write(A);
```

- (a) Write a non-serialisable schedule for these transactions. (30/100)
- (b) Add lock and unlock instructions to transactions T1 and T2 so that they observe the two-phase locking protocol. (30/100)
- (c) Write a serialisable time-stamp protocol for these transactions, indicate timestamp values for each read/write operation. (40/100)
4. (a) Assume a system having a log with immediate updates has the following log entries, ending with a system crash:

```
<R, begin>
<R, X, 1, 5>
<S, begin>
<S, X, 5, 10>
<R, Y, -1, 0>
<S, Z, 8, 12>
<Q, begin>
<S, commit>
<Q, A, -10, 10>
<Checkpoint record>
<R, commit>
<T, start>
<T, Y, 0, 15>
<Q, B, 10, 20>
<Q, commit>
----- system crash -----
```

- (i) Which transactions, if any, need to be redone? (10/100)
- (ii) Which transactions, if any, need to be undone? (10/100)
- (iii) Which transactions, if any, are not affected by the crash? (10/100)

- (b) Rewrite the log entries for the transactions of the previous task for the logging method with deferred updates. (30/100)
- (c) Describe deadlock prevention schemes. (40/100)
5. (a) Consider computer-aided architectural design as an example application area. Model a house with different rooms including walls windows and doors. Give the generalisation-specialisation hierarchy for different rooms, indicate the relationships "part-of". (50/100)
- (b) Describe the aggregation abstraction, give examples. (50/100)
6. (a) Describe different kinds of object structure, give examples. (50/100)
- (b) Define an object type Person. Use it in the definition of subtypes Employee and Student (the implementation of operations can be omitted). (50/100)
7. (a) Describe the purpose and the mechanism of object initialisers. (30/100)
- (b) Declare and implement operations and bodies of an object type "Circle" with the following operation:
- (i) initialiser Circle creating a circle using the values of two coordinates, x and y, and radius r.
  - (ii) get the x coordinate of a circle;
  - (iii) get the y coordinate of a circle;
  - (iv) get the radius of a circle;
  - (v) compute the square of a circle;
  - (vi) compare two circles for equality;
  - (vii) resize a circle according to a new radius value;
  - (viii) move a circle according to new coordinate values x and y. (70/100)

8. (a) Consider that the following relation:

**EMPLOYEE (NAME, ADDRESS, SALARY, PLANT-NUMBER)**

is fragmented vertically so that one fragment contains NAME and ADDRESS attributes and the other one - SALARY and PLANT-NUMBER attributes. The first fragment has a replica at each site. The second fragment is fragmented horizontally and each of these fragments has two replicas, one stored at the New York site and one stored locally at the plant site. Describe a good processing strategy for the following queries entered at the San Jose plant site (the result should be printed at San Jose, too):

- (i) Find the names of all employees at the Boca plant; (20/100)
- (ii) Find the address of the highest-paid employee at each of the following sites: San Jose, Montreal; (20/100)
- (iii) Find the lowest-paid employee in the company. (20/100)
- (b) Describe a possible naming scheme providing for local autonomy and network transparency in a distributed database system. (40/100)