Database Sample Examination

(Spring 2007)

Part 1: SQL

Question 1: Draw a simple ER diagram that results in a primary key/foreign key constraint to be created between the tables:

```
CREATE TABLE Salespersons (
sid CHAR(10),
primary key (sid))

CREATE TABLE Customers(
cid CHAR(10),
sid CHAR(10),
PRIMARY KEY (cid),
FOREIGN KEY (sid) REFERENCES Salespersons)
```

Question 2: Consider the following schema for an airline database:

FLIGHTS (flight_num, source_city, destination_city)
DEPARTURES (flight_num, date, plane_type)
PASSENGERS (passenger_id, passenger_name, passenger_address)
BOOKINGS (passenger_id, flight_num, date, seat_number)

The key fields are underlined. Express the following queries in SQL (feel free to abbreviate relation and attribute names and to use ***INTERSECT*** and ***EXCEPT*** if you need to).

1) Find the passenger_id of all passengers who have a seat booked on a plane of type "747" from Amsterdam to Washington. (*Do not return any duplicate values*)
2) Find the cities that have direct (non-stop) flights to both Honolulu and Newark.
3) Find the flight_num and data of all flights for which there are no reservations

Question 3: Consider the following schema:

Sailors (sid, sname, rating, age)
Boats (bid, bname, color)
Reserves (sid, bid, day)

The key fields are underlined. Please indicate whether the relational algebra expressions are equivalent to each other based on the given queries.

1) Find the names of sailor who have reserved boat 107.
   - $\pi_{sname}(\sigma_{bid=107}(\mathrm{Re}\,serves \bowtie Sailors))$
   - $\pi_{sname}((\sigma_{bid=107}\ \mathrm{Re}\,serves) \bowtie Sailors)$
2) Find the names of sailors who have reserved a blue boat.
   - $\pi_{sname}((\sigma_{color='blue'}Boats) \bowtie \mathrm{Re}\,serves \bowtie Sailors)$
   - $\pi_{sname}(\pi_{sid}((\pi_{bid}\sigma_{color='blue'}Boats) \bowtie \mathrm{Re}\,serves) \bowtie Sailors)$

3) Find the names of sailors who have reserved a red or a green boat.

- $\rho(Tempboats, (\sigma_{color='red'} Boats) \cup (\sigma_{color='green'} Boats))$
  $\pi_{sname}(Tempboats \rhd\lhd \mathrm{Re}\, serves \rhd\lhd Sailors)$

- $\rho(Tempboats, (\sigma_{color='red' \vee color='green'} Boats))$
  $\pi_{sname}(Tempboats \rhd\lhd \mathrm{Re}\, serves \rhd\lhd Sailors)$

Question 4: Given the SQL queries:

```
Q1:
SELECT a
FROM R
WHERE b >= ANY(SELECT d FROM S WHERE c>10);

Q2:
SELECT a
FROM R
WHERE b >= ALL(SELECT d FROM S WHERE c>10);
```

(A) Q1 and Q2 produce the same answer.
(B) The answer to Q1 is always contained in the answer to Q2.
(C) The answer to Q2 is always contained in the answer to Q1.
(D) Q1 and Q2 produce different answers.

Question 5: In this question, *R(x)* is the schema of relation *R*.

```
Q1:
SELECT x
FROM R rr
WHERE NOT EXISTS(
     SELECT * FROM R WHERE x > rr.x);

Q2:
SELECT MAX(x) FROM R;
```

(A) Q1 and Q2 produce the same answer.
(B) The answer to Q1 is always contained in the answer to Q2.
(C) The answer to Q2 is always contained in the answer to Q1.
(D) Q1 and Q2 produce different answers.

Part 2: Transaction Management

Question 1: Given the interleaved schedules:

Schedule 1

T_1	R(A)	R(C)			W(C)						Commit
T_2			R(C)				W(C)	R(B)	W(B)		Commit
T_3				R(C)		R(A)				W(A)	Commit

Schedule 2

T_1		R(A)		R(C)	W(A)	Commit					
T_2			R(C)				R(B)	W(B)	Commit		
T_3	R(B)									W(B)	Commit

Schedule 3

T_1	R(C)	W(A)			W(A)	Commit	
T_2			W(A)	R(B)		W(B)	Commit

1) Which of the following schedules are serializable? Give a serial schedule or identify possible anomalies.
2) Draw the precedence graph for all three schedules and check whether they are conflict-serializable or not.
3) Apply strict 2PL to the non-conflict-serializable schedules
4) In one of the schedules a deadlock emerges – draw the waits-for-graph for this schedule after all transactions are captured in the deadlock situation.

(Use X(.) to denote exclusive locks and S(.) to denote shared locks!)

Part 3: Schema Refinement

Question 1: Consider a relational schema ABCDEFGHIJ, which contains the following FDs: AB→C, D→E, AE→G, GD→H, IF→J.

1) Check whether or not the functional dependencies entail
    a) ABD→GH
    b) ABD→HJ
    c) ABC→G
    d) GD→HE

2) Let A denote a key for the aforementioned relation. Derive a lossless join, dependency preserving decomposition in 3NF!

3) Test whether or not the decomposition is also in BCNF!

4) Prove formally the following theorem by using Armstrong's axioms!

(Theorem: Any relation that satisfies X→ Y and X→ Z must also satisfy X→YZ)

Part 4: Questions

Question 1:

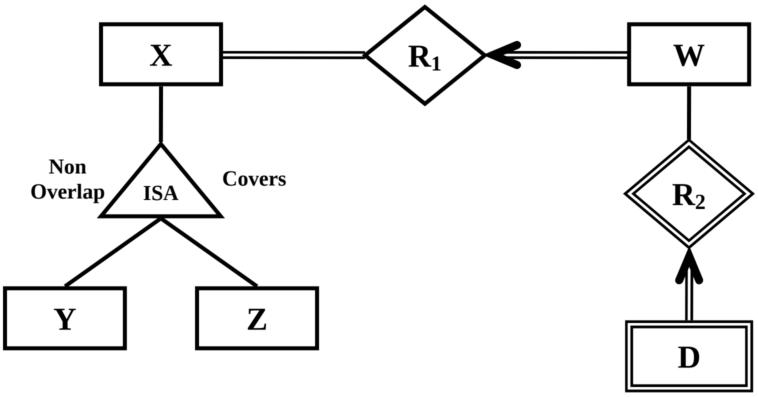

Figure: E-R Diagram

1. In the E-R diagram in shown in figure 1, the diamond labeled R_1 depicts:

   (A) An entity
   (B) An entity set
   (C) An relationship
   (D) An relationship set

2. Based on the E-R diagram shown in figure 1, find the correct description for participation constraints:

   (A) For each entity $x \in X$, there exists at least one entity $w \in W$, such that xR_1w
   (B) For each entity $x \in X$, the case, which there exists no entity $w \in W$ such that xR_1w, can happen
   (C) For each entity $x \in X$, there exists only one entity $w \in W$, such that xR_1w
   (D) For each entity $x \in X$, there exists at most one entity $w \in W$, such that xR_1w

Question 2: What do the ACID properties stand for? Give a brief description of the four characteristics.

Question 3: What are the serial schedule, equivalent schedules and serializable schedule?

Question 4: What is a file organization? What is an index? What is the relationship between files and indexes? Can we have several indexes on a single file of records?

Question 5: How is a data organized in a tree-based index? When would you use a tree-based index?

Question 6: What are/is the primary goal(s) in schema normalization? Mark the correct answer(s).

(A) Simplify the conceptual schema
(B) Reduce number of tables
(C) Avoid redundancy
(D) Make the access of data more efficient

Question 7: Describe briefly the three tiers of the three-tier-architecture! Provide at least the name of one computer language that is used for each one of these layers!