Exercise 2 “Implementation of Databases”
Due until 8 Nov, 2007 (before lecture) WS 07/08

Exercise 2.1:[Database Architecture]

1. Name each of the five layers in the database architecture specified in the lecture, explain the concepts handled in each layer, and the interfaces between layers.

2. The following tasks belong to different layers, sort them so that they match the architecture top-down.
   
   (a) buffering
   
   (b) logical relation and cursor management
   
   (c) media access
   
   (d) access path management
   
   (e) view formulation and management

3. Why is the architecture not fully implemented in DBMS?

Exercise 2.2:[Query Languages]

1. Assume that no two columns of relations have the same name. Show that a query in the normal form \( \pi_{\vec{c}}(\sigma_F(R_1 \times R_2 \times \ldots \times R_n)) \) can be expressed in Domain Relational Calculus (DRC), where \( \vec{c} \) is a vector of columns, \( F \) is a boolean formula built from conjunctions of atoms in the form \( c_i = c_j \) or \( c_i = \text{constant} \) (\( c_i \) and \( c_j \) are columns).

2. What does “relational completeness” mean? Show that SQL is relational complete by enumerating SQL constructs corresponding to selection, projection, cartesian product, union, and difference.

3. Give two examples of SQL constructs/semantics not expressible in relational algebra (RA).

4. Suppose we have three tables \( \text{Professor}(id) \), \( \text{Student}(id) \), and \( \text{Female}(id) \). Translate the following SQL query into relational calculus and relational algebra.

   ```sql
   select Female.id
   from Professor, Student, Female
   where Professor.id=Female.id or Student.id=Female.id
   ```
Figure 1: Query Flow through DBMS

5. Figure 1 shows the flow of a query through a DBMS, in which different forms are used to represent a query at different stages. Fill in the three blanks with the corresponding query languages (i.e., SQL, RC, RA).

Exercise 2.3:[Query Formulation]
Given are the following relations from a games database:

- published with attributes game and publisher, which contains the publishers of each game.
  \[\text{published} = \text{game}, \text{publisher}\]
- designed with attributes game and designer, which contains the designers of each game.
  \[\text{designed} = \text{game}, \text{designer}\]
- reviewed with attributes game and reviewer, which contains the reviewers of each game.
  \[\text{reviewed} = \text{game}, \text{reviewer}\]

Every relation represents a n:m-relationship, i.e. the key is formed by all attributes of the relation. Define the following queries as expressions in SQL, RA, and RC (first two in TRC, the other two in DRC).

1. Find all designers of the game ”Tetris”. (1 pt.)
2. Find all publishers of all games designed by ”Schmidt”. (1 pt.)
3. Find all designers who have designed more than two games. (2 pt.)
4. Find all reviewers who have reviewed at least one game that has not yet been published. (1 pt.)