CS 589 Principles of Database Systems
Fall 2008 Quiz 7 Monday, November 24, 2008 Suggested Answers

Hint: if any of the queries that you are asked to write for this quiz are impossible to write, you should explain why they are impossible to write.

1. Consider the following database schema where id is the key for the Person table, with attributes of name and age. Vehicle identification number (vin) is the key for the Car table with attributes of make (e.g., "Toyota" or "Ford"), model (e.g., "Corolla" or "Mustang"), and year. The key for the Owns table is (id, vin) together. Owns.id is a foreign key that REFERENCES Person.id; Owns.vin is a foreign key that REFERENCES Cars.vin.

   Person(<u>id</u>, name, age)
   Car(<u>vin</u>, make, model, year)
   Owns(<u>id, vin</u>)

   a. (5 points) Write a safe Datalog query (with negation, without recursion) against that schema.
   You can write any safe query that you like. Here's a simple one.
   Answer (name) :- Person (id, name, 21).
   This query finds the name of all persons who are age 21.

   b. (5 points) Write an equivalent query in allowed domain relational calculus.

   { n:name | ∃i:id (Person(i, n, 21) }

   c. (5 points) Write an equivalent query in relational algebra.

   $\Pi_{name}(\sigma_{age=21}(Person))$

2. Using the same schema as above:
   a. (5 points) Write a Datalog query (with negation, without recursion) that is NOT safe.
   In order to be safe, all variables in the query must occur positively in the body.
   Here's a query that is not safe because all three variables appears ONLY negatively in the body.

   Answer(id, name, age) :- ¬ Person(id, name, age).

   This finds all the (id, name, age) triples that do NOT appear in the database as Person tuples.

   b. (5 points) Write an equivalent query in domain relational calculus.

   This query is not allowed because i, n, and a appear only negatively in the formula. In order for a formula to be allowed, every free variables must occur positively in the formula (plus other constraints, see Definition 3.50 in the textbook).
   { i:id, n:name, a:age | ¬ Person(i, n, a) }

   c. (5 points) Write an equivalent query in relational algebra.

   It is not possible to write an equivalent relational algebra query for ANY unsafe Datalog query or any non-allowable calculus query in relational algebra.