Problem Set #3

Problem 1:

Consider a relational schema R = (A, B, C, D, E, H, I) satisfying the functional dependencies F = {A → D, DB → AC, HD → ABH, A → H, E → H, H → E}.

a) Derive all candidate keys for this schema.
b) Derive a canonical cover of the functional dependencies in F
c) Is the above schema in BCNF? Prove or disprove. If it is not in BCNF, convert it into BCNF.
d) Is the BCNF schema from c) dependency-preserving? Prove or disprove. If not, convert it into 3NF.

Problem 2:

Consider the following schema for a short-term car rental business, where all attributes are combined into just two tables:

CarRental(custid, custname, custcity, coststate, carid, carmodel, caryear, rentdate, rentalfee, citytax,
pickupbid, returnbid) /* pickupbid and returnbid are foreign keys referencing bid in Branch */
Branch(bid, bcity, bstate)

In this schema, we assume that car rentals are always for one day. Customers are identified by a custid, and have a name, city, and state of residence. Cars are identified by a carid, and have a model and a year when the car was made. For each rental, we store the rental fee that is paid to the company, as well as a city tax

that goes to the city where the car is picked up. Cars can be picked up and returned in many branches of the company, where each branch is identified by a bid. In addition, assume that there is a rule that requires that two customers who pick up the same model of car on the same day at the same branch always pay the same rental fee.

a) Explain why the above is not a good relational design. Name several reasons.
b) Identify the set F of non-trivial functional dependencies for this schema. (It is enough to identify a subset E such that the closures of E and F are the same.)
c) Derive a canonical cover of the functional dependencies in F.
d) Is the above schema in BCNF? Prove or disprove. If it is not in BCNF, convert it into BCNF. Show the steps in your decomposition and the functional dependencies you are using to decompose.
e) Is the BCNF schema from d) dependency-preserving? Prove or disprove. If not, convert it into 3NF.
f) Suppose we add an additional constraint that says that the citytax that is collected must be the same for all rentals that were picked up in the same city, but that it might be different for different cities. (E.g., maybe Houston charges a tax of $10 per rental, Chicago $5, and Dallas $0.) How would this change your answers for parts b) through e)?
g) Suppose that instead of the constraint in f), we have a constraint that all cars must be returned to a branch in the same state? E.g., if you pick up a car at a branch in Texas, you can only return it at other branches in Texas. Can you handle this sort of constraint using functional dependencies? How would it change your solution? (Discuss briefly – you do not need to go through all the steps again.)

Problem 3:

In this problem, you are asked to explore the metadata querying facilities in your database system. The answers may depend on which system you have installed, so make sure to state which system you are using! Write the following queries using the bakery database dicussed in class. Data will be provided. Please submit screenshots of the result.

a) List all tables whose name starts with "C".
b) List all foreign keys referencing oid in the Order table
c) List all tables and the number of attributes each table has.
d) For each attribute in the Order table, output how many distinct values the attribute has in the database.
e) List any customers whose name contains the name of an ingredient. (E.g., there could be customer with name "Barry Sugar", or "Frank Salt".)