EXERCISES

7.1: Suppose we have a database for an investment firm, consisting of the following attributes: B (broker), O (office of a broker), I (investor), S (stock), Q (quantity of stock owned by an investor), and D (dividend paid by a stock), with the following functional dependencies: $S \to D$, $I \to B$, $IS \to Q$, and $B \to O$.

a) Find a key for the relation scheme $R = BOSQID$.
b) How many keys does relation scheme R have? Prove your answer.
c) Find a lossless join decomposition of R into Boyce-Codd normal form.
d) Find a decomposition of R into third normal form, having a lossless join and preserving dependencies.

7.2: Suppose we choose to represent the relation scheme R of Exercise 7.1 by the two schemes $ISQD$ and IBO. What redundancies and anomalies do you forsee?

7.3: Suppose we instead represent R by SD, IB, ISQ, and BO. Does this decomposition have a lossless join?

7.4: Suppose we represent R of Exercise 7.1 by ISQ, IB, SD, and ISO. Find minimal covers for the dependencies (from Exercise 7.1) projected onto each of these relation schemes. Find a minimal cover for the union of the projected dependencies. Does this decomposition preserve dependencies?

7.5: In the database of Exercise 7.1, replace the functional dependency $S \to D$ by the multivalued dependency $S \to\to D$. That is, D now represents the dividend "history" of the stock.

a) Find the dependency basis of I.
b) Find the dependency basis of BS
c) Find a fourth normal form decomposition of R.

7.6: Consider a database of ship voyages with the following attributes: S (ship name), T (type of ship), V (voyage identifier), C (cargo carried by one ship on one voyage), P (port), and D (day). We assume that a voyage consists of a sequence of events where one ship picks up a single cargo, and delivers it to a sequence of ports. A ship can visit only one port in a single day. Thus, the following functional dependencies may be assumed: $S \to T$, $V \to SC$, and $SD \to PV$.

a) Find a lossless-join decomposition into BCNF.
b) Find a lossless-join, dependency-preserving decomposition into 3NF.
* c) Explain why there is no lossless-join, dependency-preserving BNCF decomposition for this database.

7.7: Let U be a set of attributes and D a set of dependencies (of any type) on the attributes of U. Define SAT(D) to be the set of relations r over U such that r satisfies each dependency in D. Show the following.

a) $\text{SAT}(D_1 \cup D_2) = \text{SAT}(D_1) \cap \text{SAT}(D_2)$.
b) If D_1 logically implies all the dependencies in D_2, then

$$\text{SAT}(D_1) \supseteq \text{SAT}(D_2)$$

7.8: Complete the proof of Lemma 7.1; i.e., show that the transitivity axiom for functional dependencies is sound.

7.9: Complete the proof of Theorem 7.2 by showing statement (*):

$$\text{If } X_1 \subseteq X_2 \text{ then } X_1^{(j)} \subseteq X_2^{(j)} \text{ for all } j$$

7.10: Let F be a set of functional dependencies.

a) Show that $X \to A$ in F is redundant if and only if X^+ contains A, when the closure is computed with respect to $F - \{X \to A\}$.
b) Show that attribute B in the left side X of a functional dependency $X \to A$ is redundant if and only if A is in $(X - \{B\})^+$, when the closure is taken with respect to F.

* **7.11:** Show that singleton left sides are insufficient for functional dependencies. That is, show there is some functional dependency that is not equivalent to any set of functional dependencies $\{A_1 \to B_1, \ldots, A_k \to B_k\}$, where the A's and B's are single attributes.

* **7.12:** Develop the theory of functional dependencies with single attributes on the left and right sides (call them *SAFD's*). That is:

a) Give a set of axioms for SAFD's; show that your axioms are sound and complete.
b) Give an algorithm for deciding whether a set of SAFD's implies another SAFD.
c) Give an algorithm to test whether two sets of SAFD's are equivalent.
d) SAFD's look like a familiar mathematical model. Which?

* **7.13:** In Theorem 7.3 we used two transformations on sets of functional dependencies to obtain a minimal cover:

i) Eliminate a redundant dependency.
ii) Eliminate a redundant attribute from a left side.

Show the following:

a) If we first apply (*ii*) until no more applications are possible and then apply (*i*) until no more applications are possible, we always obtain a minimal cover.

b) If we apply first (i) until no longer possible, then apply (ii) until no longer possible, we do not necessarily reach a minimal cover.

7.14: A relation scheme R is said to be in *second normal form* if whenever $X \rightarrow A$ is a dependency that holds in R, and A is not in X, then either A is prime or X is not a proper subset of any key (the possibility that X is neither a subset nor a superset of any key is not ruled out by second normal form). Show that the relation scheme $SAIP$ from Example 7.14 violates second normal form.

7.15: Show that if a relation scheme is in third normal form, then it is in second normal form.

7.16: Consider the relation scheme with attributes S (store), D (department), I (item), and M (manager), with functional dependencies $SI \rightarrow D$ and $SD \rightarrow M$.

a) Find all keys for $SDIM$.

b) Show that $SDIM$ is in second normal form but not third normal form.

* 7.17: Give an $O(n)$ algorithm for computing X^+, where X is a set of at most n attributes, with respect to a set of functional dependencies that require no more than n characters, when written down.

* 7.18: Complete the proof of Theorem 7.5 by providing a formal proof that in the row for R_1, an a is entered if and only if $R_1 \cap R_2 \rightarrow A$.

7.19: Complete the proof of Lemma 7.5 by showing that if $r \subseteq s$ then

$$\pi_{R_i}(r) \subseteq \pi_{R_i}(s)$$

7.20: In Example 7.10 we contended that $Z \rightarrow C$ does not imply $CS \rightarrow Z$. Prove this contention.

7.21: At the end of Section 7.5 it was claimed that $\rho = (AB, CD)$ was a dependency-preserving, but not lossless-join decomposition of $ABCD$, given the dependencies $A \rightarrow B$ and $C \rightarrow D$. Verify this claim.

7.22: Let $F = \{AB \rightarrow C,\ A \rightarrow D,\ BD \rightarrow C\}$.

a) Find a minimal cover for F.

b) Give a 3NF, dependency-preserving decomposition of $ABCD$ into only two schemes (with respect to the set of functional dependencies F).

c) What are the projected dependencies for each of your schemes?

d) Does your answer to (b) have a lossless join? If not, how could you modify the database scheme to have a lossless join and still preserve dependencies?

7.23: Let $F = \{AB \rightarrow C,\ A \rightarrow B\}$.

a) Find a minimal cover for F.

b) When (a) was given on an exam at a large western university, more than half the class answered $G = \{A \rightarrow B,\ B \rightarrow C\}$. Show that answer is wrong by giving a relation that satisfies F but violates G.

7.24: Suppose we are given relation scheme $ABCD$ with functional dependencies $\{A \rightarrow B,\ B \rightarrow C,\ A \rightarrow D,\ D \rightarrow C\}$. Let ρ be the decomposition (AB, AC, BD).

a) Find the projected dependencies for each of the relation schemes of ρ.

b) Does ρ have a lossless join with respect to the given dependencies?

c) Does ρ preserve the given dependencies?

7.25: Show that (AB, ACD, BCD) is not a lossless-join decomposition of $ABCD$ with respect to the functional dependencies $\{A \rightarrow C,\ D \rightarrow C,\ BD \rightarrow A\}$.

7.26: Consider the relation scheme $ABCD$ with dependencies

$$F = \{A \rightarrow B,\ B \rightarrow C,\ D \rightarrow B\}$$

We wish to find a lossless-join decomposition into BCNF.

a) Suppose we choose, as our first step, to decompose $ABCD$ into ACD and BD. What are the projected dependencies in these two schemes?

b) Are these schemes in BNCF? If not, what further decomposition is necessary?

7.27: For different sets of assumed dependencies, the decomposition

$$\rho = (AB, BC, CD)$$

may or may not have a lossless join. For each of the following sets of dependencies, either prove the join is lossless or give a counterexample relation to show it is not.

a) $\{A \rightarrow B,\ B \rightarrow C\}$.

b) $\{B \rightarrow C,\ C \rightarrow D\}$.

c) $\{B \twoheadrightarrow C\}$.

* 7.28: At most how many passes does Algorithm 7.3 (the test for dependency-preservation) need if F is a set of n functional dependencies over m attributes (an order-of-magnitude estimate is sufficient).

* 7.29: Let F be a set of functional dependencies with singleton right sides.

a) Show that if a relation scheme R has a BCNF violation $X \rightarrow A$, where $X \rightarrow A$ is in F^+, then there is some $Y \rightarrow B$ in F itself such that $Y \rightarrow B$ is a BCNF violation for R.

b) Show the same for third normal form.

7.30: Show the following observation, which is needed in Theorem 7.8. If R is a relation scheme, and $X \subseteq R$ is a key for R with respect to set of functional dependencies F, then X cannot have a 3NF violation with respect to the set of dependencies $\pi_X(F)$.

7.31: Prove that there is no such thing as an "embedded functional dependency." That is, if $S \subseteq R$, and $X \rightarrow Y$ holds in $\pi_S(R)$, then $X \rightarrow Y$ holds in R.

* 7.32: Complete the proof of Theorem 7.9 by showing that axioms A1–A8 are sound and complete. *Hint*: The completeness proof follows Theorem 7.1. To find a counterexample relation for $X \twoheadrightarrow Y$, we generally need more than a two-tuple relation as was used for functional dependencies; the relation could have 2^b tuples, if b is the number of blocks in the dependency basis for X.

* 7.33: Verify the union, pseudotransitivity, and decomposition rules for multivalued dependencies.

* 7.34: Verify the contention in Example 7.21, that there is a relation r satisfying $SP \rightarrow Y$, such that $\pi_{CS}(r) \bowtie \pi_{CP}(r) \bowtie \pi_{SPY}(r) \neq r$. Check that your relation does not satisfy $C \twoheadrightarrow S \mid P$.

7.35: Given the dependencies $\{A \twoheadrightarrow B,\ C \rightarrow B\}$, what other nontrivial multivalued and functional dependencies hold over the set of attributes ABC?

* 7.36: Prove that in $ABCD$ we can infer $A \twoheadrightarrow D$ from $\{A \twoheadrightarrow B,\ A \rightarrow C\}$ in each of the following ways.

a) Directly from the definitions of functional and multivalued dependencies.

b) From axioms A1–A8.

c) By converting to generalized dependencies and "chasing."

* 7.37: Near the beginning of Section 7.10 we claimed that we could project a set of multivalued and functional dependencies D onto a set of attributes S by the following rules (somewhat restated).

i) $X \rightarrow Y$ is in $\pi_S(D)$ if and only if $XY \subseteq S$ and $X \rightarrow Y$ is in D^+.

ii) $X \twoheadrightarrow Y$ is in $\pi_S(D)$ if and only if $X \subseteq S$, and there is some multivalued dependency $X \twoheadrightarrow Z$ in D^+, such that $Y = Z \cap S$.

Prove this contention.

7.38: Show that the decomposition (CHR, CT, CSG) obtained in Example 7.20 is not lossless with respect the the given functional dependencies only; i.e., the multivalued dependency $C \twoheadrightarrow HR$ is essential to prove the lossless join.

7.39: Use the chase algorithm to tell whether the following inferences are valid over the set of attributes $ABCD$.

a) $\{A \twoheadrightarrow B, A \rightarrow C\} \models A \twoheadrightarrow D$

b) $\{A \twoheadrightarrow B \mid C, B \twoheadrightarrow C \mid D\} \models A \twoheadrightarrow C \mid D$

c) $\{A \twoheadrightarrow B \mid C, A \rightarrow D\} \models A \twoheadrightarrow C \mid D$

** d) $\{A \twoheadrightarrow B \mid C, A \twoheadrightarrow C \mid D\} \models A \twoheadrightarrow B \mid D$

* 7.40: Show that no collection of tuple-generating dependencies can imply an equality-generating dependency.

7.41: State an algorithm to determine, given a collection of functional, (full) multivalued, and (full) join dependencies, whether a given decomposition has a lossless join.

7.42: Show that the multivalued dependency $X \twoheadrightarrow Y$ over the set of attributes U is equivalent to the join dependency $\bowtie(XY, XZ)$, where $Z = U - X - Y$. *Hint*: Write both as generalized dependencies.

7.43: What symbol mapping explains the application of Figure 7.11(b) to Figure 7.12(b) to deduce Figure 7.12(c)?

* 7.44: Show that Theorem 7.11, stated for functional and multivalued dependencies, really holds for arbitrary generalized dependencies. That is, (R_1, R_2) has a lossless join with respect to a set of generalized dependencies D if and only if $(R_1 \cap R_2) \twoheadrightarrow (R_1 - R_2)$.

* 7.45: Show that if decomposition $\rho = (R_1, \ldots, R_k)$ has a lossless join with respect to a set of generalized dependencies D, then the decomposition $(R_1, \ldots, R_k, S)$ also has a lossless join with respect to D, where S is an arbitrary relation scheme over the same set of attributes as ρ.

* 7.46 Show that it is $\mathcal{NP}$-hard ($\mathcal{NP}$-complete or harder—see Garey and Johnson [1979]) to determine:

a) Given a relation scheme R and a set of functional dependencies F on the attributes of R, whether R has a key of size k or less with respect to F?

b) Given R and F as in (a), and given a subset $S \subseteq R$, is S in BNCF with respect to F?

c) Whether a given set of multivalued dependencies implies a given join dependency.

* 7.47: A *unary inclusion dependency* $A \subseteq B$, where A and B are attributes (perhaps from different relations) says that in any legal values of the relation(s), every value that appears in the column for A also appears in the column for B. Show that the following axioms

i) $A \subseteq A$ for all A.

ii) If $A \subseteq B$ and $B \subseteq C$ then $A \subseteq C$.