CSCI 403 - Database Management
Sample Quiz 3

Instructions:

Circle **one** answer for each question.

Questions:

These questions concern the relation schema and functional dependencies described below.

Relation schema `fruit`:

Attributes: {apple, banana, cherry, date, elderberry, fig}

Key: {apple, banana}

Functional dependencies:
{apple, banana} → {cherry, date, elderberry, fig}
{apple} → {cherry}
{apple} → {date}
{date} → {elderberry}
{fig} → {date}

1. This relation schema is:

   (a) In Boyce-Codd Normal Form (BCNF).

   (b) Is not in BCNF because it has a composite key.

   (c) Is not in BCNF because it has a functional dependency {apple} → {cherry}, and {apple} is not a superkey.

   (d) Is not in BCNF because it has a functional dependency {apple} → {cherry}, and {cherry} is not a superkey.

   Answer: c

2. The set {apple, banana, cherry}:

   (a) Is a superkey.

   (b) Is functionally determined by {apple, banana}.

   (c) Functionally determines {fig}.

   (d) All of the above.

   Answer: d

3. Which of these are functional dependencies that can be inferred from the functional dependencies provided?

   (a) {apple} → {elderberry}.
   (b) {date} → {fig}.
   (c) {banana} → {cherry, date}.
   (d) All of the above.

   Answer: a

4. What is the closure of {apple}?

   (a) {apple, cherry, date}.
   (b) {apple, cherry, date, elderberry}.
   (c) {apple, banana, cherry, date, elderberry}.
   (d) {apple, banana, cherry, date, elderberry, fig}.

   Answer: b

5. Which of these functional dependencies violate BCNF?

   (a) {apple, banana} → {cherry, date, elderberry, fig}.
   (b) {apple} → {cherry}.
   (c) {date} → {elderberry}.
   (d) Both (b) and (c).

   Answer: d

6. Which of the following would be a decomposition of `fruit` that moves the schema closer to BCNF?

   (a) R1 = {apple, cherry, date, fig}, R2 = {apple, banana, date, fig}.
   (b) R1 = {apple, banana}, R2 = {cherry, date, elderberry, fig}.
   (c) R1 = {date, fig}, R2 = {apple, banana, cherry, elderberry, fig}.
   (d) R1 = {apple, banana, cherry, date}, R2 = {apple, banana, elderberry, fig}.

   Answer: c

   Notes:

   This is obviously a completely abstract example, so you have to work out the answers in a purely mathematical way, using the rules discussed in class.

   Note that the attributes have first letters starting with sequential letters of the alphabet. This is so, if it helps, you can save time by just writing notes to yourself about a, b, c, etc., rather than writing out the whole attribute name.

   The last question is, in my opinion, the hardest. Here you should consider using a process of elimination. I.e., can you see why the incorrect answers do not constitute a good decomposition according to the algorithm we learned? Here's my take on them:

   · (a) Not all of the attributes are included in the decomposition products.

   · (b) There is no overlapping attribute in the two products; i.e., there is no way to re-join them to recover the original relation schema.

   · (d) The only attributes in common between the two products are apple and banana, which implies we did a decomposition where the left-hand side is {apple, banana}. However, there is no reason to decompose on these attributes, as they constitute a key (and therefore superkey) of the original schema.