Instructions: Turn in your answers to the questions below in class on or before the due date. You must include at least a brief summary of each question below as part of your answer (including the question number). Be sure your answers are legible. Note that each question is worth 1 point (for a total of 10 points).

1. Give an example of a relation (different from what we have done in class) that contains a primary key along with two non-key, non-trivial functional dependencies. (a) State the functional dependencies; (b) give an instance that contains redundant data; and (c) give an example of each type of anomaly that can occur (update, insertion, and deletion).

2. Given a set of functional dependencies (FDs), we can use the following rules to infer new dependencies implied by the set.

   • Reflexivity: If $Y \subseteq X$, then $X \rightarrow Y$
   • Augmentation: if $Z \subseteq W$ and $X \rightarrow Y$, then $XW \rightarrow YZ$
   • Transitivity: If $X \rightarrow Y$ and $Y \rightarrow Z$, then $X \rightarrow Z$

Using these rules, answer the following questions:

(a). Show that if $X \rightarrow Y$ and $YW \rightarrow Z$, then $XW \rightarrow Z$.

(b). Show that if $X \rightarrow YZ$, then $X \rightarrow Y$ and $X \rightarrow Z$.

(c). Show that if $X \rightarrow Y$ and $X \rightarrow Z$, then $X \rightarrow YZ$.

3. FDs can be used to determine a relation’s candidate keys. In particular, a candidate key is a minimal set of attributes that all other attributes functionally depend on. We can test if a set of attributes is a superkey by repeatedly applying the rules in question 1. For example, given the relation $R(a, b, c)$ and the FDs $a \rightarrow b$ and $b \rightarrow c$, we can apply transitivity to determine that $a \rightarrow c$, reflexivity to determine that $a \rightarrow a$, and $a \rightarrow b$ is given. Thus $a$ is a candidate key since it is minimal and since $a \rightarrow abc$. Using a similar approach, find the candidate keys for the following.

(a). A relation $R(a, b, c)$ and FDs $\{a \rightarrow c\}$

(b). A relation $R(a, b, c, d)$ and FDs $\{b \rightarrow c, d \rightarrow a\}$

(c). A relation $R(a, b, c, d)$ and FDs $\{a \rightarrow c, c \rightarrow d\}$

(d). A relation $R(a, b, c, d, e)$ and FDs $\{c \rightarrow b, bd \rightarrow e, a \rightarrow d, e \rightarrow a\}$

4. Assume we have the following (at least 1NF) relational schema:

   movie_info(title, director, studio, studio_loc, award_type, award_year)

with functional dependencies

   award_type, award_year $\rightarrow$ title, director
For each of the following decomposed movie_info tables, state the candidate key(s) for the table and the highest normal form each table is in (i.e., either 1NF, 2NF, 3NF, or BCNF).

(a). movie_info_1(director, studio, studio_loc)
(b). movie_info_2(title, director, studio)
(c). movie_info_3(award_type, award_year, title)
(d). movie_info_4(award_type, award_year, studio_loc)
(e). movie_info_5(director, studio)

5. Give a lossless BCNF decomposition of the full movie_info relation. Draw the decomposition as a “tree”, as we’ve done in class, to show your work. Include the candidate keys of the final decomposed relations.

6. Give a “realistic” example (not from class) of a relation that is in 3NF but not in BCNF. To be “realistic,” the relation should be based on actual things, not just symbols. Your example cannot be from class. Show that a BCNF decomposition of your relation would lose FDs (i.e., it would not be lossless).

7. Draw an Entity-Relationship Diagram (ERD) for the following requirements.

- Each coffee roaster is identified by their company name. In addition, each coffee roaster has a founding year, a headquarters location (city and state), and a founding roaster (first and last name).
- Each coffee roaster produces zero or more coffee blends. Similarly, a blend can be produced by zero or more roasters. Roasters will typically assign their own name to a blend (e.g., “Morning Blend”).
- Each coffee blend has a unique type name (e.g., “Mocha Java”, “Espresso”). In addition, each blend has an acidity level, a flavor, the type of roast (e.g., “light”, “medium”, “dark”), and a description.

8. Draw an Entity-Relationship Diagram (ERD) for the following requirements.

- Each ski resort is identified by a unique name (e.g., “Schweitzer Mountain”, “Mt. Bachelor”). Ski resorts also consist of a location (county and state) and total elevation.
- Ski resorts consist of zero or more ski runs and ski lifts. Assume all ski lifts and ski runs have unique names. Ski runs also consist of a level (e.g., beginner, intermediate, etc.), and a total length. Ski lifts also consist of their capacity and the type of chair (e.g., double, triple, quad, etc.)
• Each ski lift can serve multiple ski runs, where a lift serves a run if you can easily get to the top of the run from the top of the lift. There can be cases where a lift doesn’t serve any runs (although this is rare). A ski run can be served by many lifts. There are cases where a run isn’t served by a lift (instead, the top of the run is served from other runs, which does not need to be modeled in current the design).

• Each ski run can connect to zero or more ski lifts, where each such lift can be entered from the bottom of the run (without having to take any other runs or lifts). You can assume that some lifts do not have a run that connects to it (although this would be rare). Note that just because a lift serves a run does not mean the run connects to the lift, and vice versa.

9. Convert your ERD from Question 7 to a relational schema using the translation approach described in class. Use the \( R(x, y, z) \) notation to describe each relations schema and include both keys (underlined), foreign keys, and any other constraints. For foreign keys, you must state what table each foreign key references.

10. Convert your ERD from Question 8 to a relational schema using the translation approach described in class. Use the \( R(x, y, z) \) notation to describe each relations schema and include both keys (underlined), foreign keys, and any other constraints. For foreign keys, you must state what table each foreign key references.