Nate Bender
CS342 – Guide 04

1. **Read Chapter 14 (focus primarily on Sections 14.1-2, 14.3.1-3, 14.5-6) — It's ok to submit rather fuzzy or even empty answers on the material specifically marked for Wednesday's discussion.**
    a. **Explain the four informal guidelines discussed in Section 14.1.**

All semantics in the schema must be clear. In other words, someone should be able to look at the schema and determine what the database is for and how it works. All redundant information should be reduced so that the database isn't any larger than it needs to be. All NULL values should be reduced such that the rows of each table is useful. Spurious tuples should also be disallowed such that the tables are all consistent and don't contain false or useless information.

    b. **Explain the terms *functional dependency* (for Monday, see Section 14.2) and *multi-valued dependency* (for Wednesday, see Section 14.6) by giving an example of each and showing how it satisfies the term's formal definition.**

Functional dependencies are defined by the constraining domain R, where x -> y can only be a certain set of tuples. Much like a mathematical function where the codomain can only be a result of the domain. For example, and employees name can only be a result of their Social Security number,

    c. **Review the formal definitions of these terms: *superkey*; *key* (see Section 14.3.3).**

Superkeys define the uniqueness constraint such that no two tuples can have the same value. It is a combination off all attributes in a row.

A key is a superkey with the additional property that removing one of its attributes makes it no longer a superkey.

    d. **Explain *BCNF* (for Monday, see Section 14.5) and *4NF* (for Wednesday, see Section 14.6) by giving an example relation and proving whether its schema conforms (or fails to conform) to the definitions of each normal form.**

Boyce-Codd normal form is a pre-existing database form from which other databases are designed. This results in high quality databases and meets all qualifications for being a good database. BCNF is defined by Y always being a subset of X in the form X -> Y, as well as X being a superkey.

    e. **Explain when one would one want to *normalize* a relational schema and when one might want to *de-normalize* one.**

De-normalization is the process of storing the join of higher normal form relations as a base relation, which is in a lower normal form. This may save read/compute times as the joins would already be made, rather than computing them from two separate tables.