CS34800 Fall 2016 Midterm 2, 9 November, 2016
Prof. Chris Clifton

Turn Off Your Cell Phone. Use of any electronic device during the test is prohibited. As previously noted, you are allowed notes: Up to two sheets of 8.5x11 or A4 paper, single-sided (or one sheet double-sided).

Time will be tight. If you spend more than the recommended time on any question, **go on to the next one**. If you can't answer it in the recommended time, you are either giving too much detail or the question is material you don't know well. You can skip one or two parts and still demonstrate what I believe to be an A-level understanding of the material.

Note: It is okay to abbreviate in your answers, as long as the abbreviations are unambiguous and reasonably obvious.

This was a tough exam. I would expect an A student to score 24 or better, a B student to score at least 17, and a C student to score at least 10.

1 Functional Dependencies and Normalization (32 minutes, 20 points)

Given the following database schema (two or three tables):
Relation X: $A \quad B \quad C$
Relation Y: $A \quad D$
Relation Z: $C \quad E \quad F$
and the following functional dependencies: $A \rightarrow B \quad BC \rightarrow A \quad AB \rightarrow D \quad CE \rightarrow F$

A. (7 minutes, 5 points) Give SQL code to create the tables. The tables should enforce as many of the functional dependencies as possible. There may be information you have to "guess" to do this, briefly explain what it is.

Need to assume data types. I'll make them all single characters.

create table X (A char(1) references Y(A),
B char(1), C char(1), primary key(B, C));

create table Y (A char(1) primary key, D char(1));

create table Z (C char(1), E char(1), F char(1),
primary key(C, E));

Note that we don't enforce $A \rightarrow B$, but can check it in a single relation (3NF). The foreign key constraint follows from assuming that we started with a single relation and normalized it. But from the information you were given, they don't necessarily hold, so no points either way...

Scoring: 1 for valid create table and table names, 1 for getting attributes, 1 for primary key constraint, 1 for noting data types needed, 1 for getting all primary keys, 1 for noting $A \rightarrow B$ isn't enforced, maximum 5 points

B. (3 minutes, 2 points) Give a minimal set for the functional dependencies.

$A \rightarrow BD$, $BC \rightarrow A$, $CE \rightarrow F$

Scoring: 1 for not expanding, 1 for getting rid of $B \rightarrow D$

C. (3 minutes, 2 points) Give the closure of the set of functional dependencies.

Ignoring any trivial FDs (i.e., everything on the LHS), $A^+ = BD$, $AB^+ = D$, $AC^+ = BD$, $BC^+ = AD$, $CE^+ = F$, $ABC^+ = D$, $ACE^+ = BDF$, $BCE^+ = ADF$, $ABCE^+ = DF$, $ACDE^+ = BF$, $ABCDE^+ = F$, $ABCEF^+ = D$, $ACDEF^+ = B$

Scoring: 1 for expanding some, 1 for reasonably complete with no extras.

D. (8 minutes, 5 points) What normal forms do the above relations satisfy (yes or no to each)? If no, say what table violates that normal form and why.

- 1NF

  Yes: each entry in a relation is a single value.
- BCNF

  No: In relation X, $A \rightarrow B$ and A is not a superkey (since $A \rightarrow C$ doesn't hold.)
- 3NF

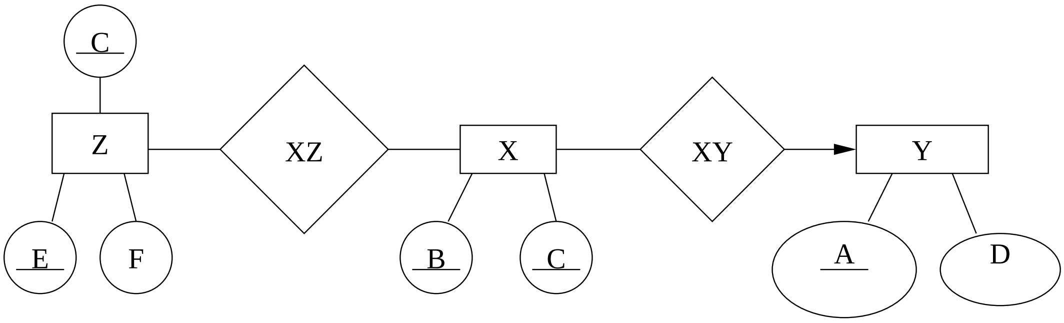
  Yes: Even though not in BCNF, in the violating FD $A \rightarrow B$, A is dependent on the candidate key BC.
- 4NF

  No: 4NF is essentially BCNF plus handling multivalued dependencies. Since we fail to satisfy BCNF, we fail 4NF. But in I'll accept answers consistent with your answer to either BCNF or 3NF, because the cases where you can satisfy 3NF and not 4NF are rare and quite subtle.

Scoring: 1 for each right (plus a bonus for getting all right), 1 point for each good explanation. With 3NF and 4NF, agreement is good for one even if you get both wrong.

E. (7 minutes, 3 points) Draw an Entity-Relationship diagram corresponding to the above schema. Capture as much of the information in the schema and functional dependencies as you can.

The following isn't perfect, but gives a good idea (and would have received a full score.)



Scoring: I didn't expect anything perfect, but gave 1 for proper entities, 1 for proper relationships (should match your foreign key constraints if you have them), 1 for keys, 1 capturing an FD using weak entity sets or one-many relationships. -1 for having egregious errors in otherwise decent solutions.

F. (4 minutes, 3 points) One of the application developers working with your database complains that he finds the normalized version hard to work with, and that the information isn't sufficient. He wants you to redesign the database to give him the following table to query:

ABCD

Briefly describe how you would respond to his request. For full credit, give SQL code to address the problem.

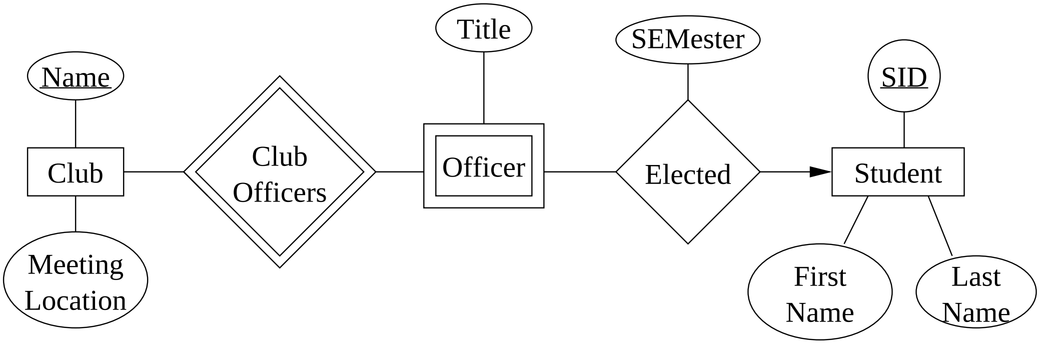
Create a view - since there isn't a need to update, this should give what the developer wants while still retaining the advantage of normalization.

```
create view ABCD as
select X.A, B, C, D
from X, Y
where X.A = Y.A;
```

Scoring: 1 for view, 1 for reasonable discussion or SQL, 1 for correct SQL

2 Database Design (17 minutes, 10 points)

Given the following Entity-Relationship diagram:



A. (7 minutes, 5 points) Give a set of relations that correspond to the E-R diagram

Club(Name, Meeting Location)
Officer(Name, Title)
Student(SID, First Name, Last Name)
Elected(Name, Title, SEMester, SID)

Scoring: 1 for relations corresponding to entities, 1 for attributes, 1 for keys (close), 1 for relation corresponding to Elected, 1 for not having weak relation.

B. (5 minutes, 3 points) Write down functional dependencies that are implied by the E-R diagram

$N \rightarrow ML$**,** $SID \rightarrow FN, LN$**,** $Name, Title \rightarrow SID$

Scoring: 1 each ($Name, Title \rightarrow SEMokay$*)*

C. (5 minutes, 2 points) Is your answer to 2A in 3rd Normal Form? If not, show the changes needed to get to 3rd Normal Form.

Yes: For each FD, the left side is a key for the relation containing all the attributes in the FD.

Scoring: 2 for correct yes, 1 for correct no, 1 for correct 3NF