

**UNCA CSCI 343**  
**Exam 3 Spring 2014**  
28 April, 2014

This is a closed book and closed notes exam. It is a 37.5 minute exam to be turned in by 11:15 AM.

Communication with anyone other than the instructor is not allowed during exams. Furthermore, calculators, PDA's, cell phones, and any other electronic or communication devices may not be used during exams.

To ensure compliance with these rules, anyone needing a break during exams must leave their exam with the instructor. The only allowed break excursion is a direct walk to and from the classroom and the restrooms outside RRO 239 (the ATMS classroom). Cell phones or computers may not be used during breaks.

Violation of these rules will result in a grade of 0 for this exam.

Name: \_\_\_\_\_

*This is a 50-point exam.*

**Problem 1 (8 points)**

Use relational algebra, with the operators  $\Pi$  for projection,  $\sigma$  for selection, and  $\times$  for Cartesian product, to encode the following SQL query.

```
SELECT H.name  
FROM   Friend F, Highschooler H  
WHERE  H.ID = F.ID1 and H.ID+1 = F.ID2 ;
```

**Problem 2 (8 points)**

Write an SQL query to list the number of students in each grade named Alison. Order your list by the grade number.

Write your answers to the remaining problems on lined paper provided by the instructor.

### Problems 3 to 6: a modification of the social relations database

We've been ordered to make some changes to the social relations database. The high school principal wants to replace crushes and friendships with clubs. His first edict is that we delete the Likes and Friend tables. Consider it done.

Now we need to restructure the database to store information about clubs. The principal has imposed some rules about clubs that our application *must* follow:

- Students can belong to several clubs.
- Students don't have to belong to any club.
- Every club must have a club president.
- The club president must be a student.
- No student may be president of more than one club.

The principal has specified some attribute names for this application. Use ClubName for the name of the club and ClubPresID for the student ID of the club's president. Retain the attribute names of the old Highschooler table: ID, Name, and grade.

#### **Problem 3 (8 points)**

Write some appropriate functional dependencies for the five attributes: ID, Name, grade, ClubName and ClubPresID. Follow the rules of the principal.

#### **Problem 4 (8 points)**

Draw either an UML or ER diagram for this database application. Keep it simple. The principal doesn't want to see a tangle of boxes and lines.

#### **Problem 5 (8 points)**

The principal likes spreadsheets. At first he suggests our database to be a single ~~table~~ table with all five attributes, as in:

ClubbyHighschooler(ID, Name, grade, ClubName, ClubPres)

However, he quickly realizes that a one table implementation can't be normal. (And every principal loves normalcy.) How would you decompose ClubbyHighschooler into tables that are in Boyce-Codd normal form?

A table is in Boyce-Codd Normal Form if for every dependency  $X \rightarrow Y$  either  $X$  is a *superkey* or  $Y$  is a subset of  $X$ .

Give some justification for your design.

#### **Problem 6 (8 points)**

Finally, write SQL CREATE TABLE statements to create the tables you described in the last problem. When appropriate include PRIMARY KEY and FOREIGN KEY constraints.

#### **Problem 7 (2 points)**

Why do many noSQL-based databases store information in a JSON-like or an XML-like structure while no SQL-based databases do? A sentence or two should do.