CS432-CS532 Homework 3
(Due: October 16, 2014 in class)

1. The following are some of the relations transformed from the ER diagram for the Student Registration System (note that some changes have been made due to the creation of a single-attribute key for Classes):

   Students(<u>sid</u>, firstname, lastname, status, gpa, email)
   Courses(<u>dept_code, course#</u>, title, credits, deptname)
   Classes(<u>classid</u>, dept_code, course#, sect#, year, semester, start_time, end_time, limit, size, classroom, capacity, fid) /* note: classid is added to serve as a single-attribute key */
   Faculty(<u>fid</u>, firstname, lastname, rank, office, email, deptname)
   Enrollments(<u>sid, classid,</u> lgrade, ngrade)

   Do the following for each relation schema:

   (a) (20 points) Identify all non-trivial functional dependencies. Don't make unrealistic assumptions about the data. Use the union rule to combine the functional dependencies as much as possible. Furthermore, if a functional dependency is redundant (i.e., it can be derived from the ones you keep), it should not be included.

   (b) (20 points) Determine whether or not the schema is in 3NF or in BCNF. Justify your conclusion.

   (c) (20) For each schema that is not in 3NF, decompose it into 3NF schemas using Algorithm LLJD-DPD-3NF. Show the result after each step of the algorithm. Are the decomposed schemas in BCNF? Justify your answer.

2. (14 points) Prove or disprove the following rules:

   (a) {B → CD, AB → E, E → C} |= {AE → CD}
   (b) {B → CD, AD → E} |= {AB → E}

   When proving a rule, you can use all the six inference rules (i.e., reflexivity rule, augmentation rule, transitivity rule, decomposition rule, union rule and pseudotransitivity rule). To disprove a rule, construct a relation with appropriate attributes and tuples such that the tuples of the relation satisfy the functional dependencies on the left of the rule but do not satisfy the functional dependency on the right of the rule.

3. (26 points) Write relational algebra expressions to answer the following query statements based on the tables in HW3_tables 1. For each query, **show the result** that would be obtained if the query were actually executed against the tables. It is ok to answer a query in multiple steps. You need to make sure that your relational algebra expressions are reasonably optimized as we discussed in class, i.e., conditions involving a single table should be specified on that table directly and Cartesian products should be replaced by joins whenever possible.

(a) (6 points) Find the dept_code, course# and title of each course that was offered in the Spring semester of 2014.

(b) (6 points) Find the first name of each student who has taken at least one CS course and at least one Math course.

(c) (7 points) Find the dept_code and course# of each course that was not offered in 2013.

(d) (7 points) Find the sid and first name of every student who has taken all CS classes offered in Spring 2014.