Comprehensive Exam
Databases
Spring 2010

1) (15 points) Circle T or F for each of the following.

— "SQL" stands for Structured Query Language. T F

— An advantage of SQL over other query languages is that SQL does not have transactions. T F

— SQL was first proposed by IBM in early 2001. T F

— One purpose of a query optimizer is to improve query performance. T F

— Every superkey is also a candidate key. T F

— Suppose a one-to-one relationship set R exists between two entity sets E1 and E2. In addition, let K1 and K2 be candidate keys for E1 and E2, respectively. Then the union of K1 and K2 forms a super key for R. T F

— The *intersection* operation is not one of the six basic relational algebraic operators because it is too inefficient. T F

— All relational algebraic expressions are "safe," by definition. T F

— The *natural join* of a relation with itself will result in an empty relation. T F

— The closure of a set of attributes can be computed using Armstrong's axioms. T F

— Armstrong's axioms include *reflexivity*, *transitivity* and *union*. T F

— Computing the closure of a set of attributes eliminates attributes from that set. T F

— The decomposition rule states that if AC->B and A->B and C->B. T F

— If a set of relational schemes is in BCNF, then that set also has a loss-less join. T F

— Let R be any relational scheme that is not in BCNF. Then R can always be decomposed into a collection of BCNF relational schemes that has a loss-less join. T F

2) (30 points) Consider the following relational schemes. Note that attributes forming the primary key for each relation have been underlined.

student(student-name, street, city)	-- Basic student information.
offering(department, number, population)	-- Courses currently offered; for CSE5260 department is "CSE" and number is 5260. Population is the number of students.
titles(department, number, title)	-- Course titles; "CSE5260" is "Database Systems"
enrollment(student-name, department, number)	-- Indicates which students are enrolled in which courses.

Give an SQL query for part (a).

(a) For each department, list the course that has the largest population. Include the department, course number, title, and population in the result.

department	number	title	population
MTH	1001	Calculus	57
CSE	4020	Database Systems	25
PSY	4260	Child Development	62
CHM	5264	Introduction to Chemistry	43

Give a relational algebraic expression for part (b).

(b) A list of the names of students who are currently enrolled in every class that "Jones" is currently enrolled in, and in every class that "Smith" is currently enrolled in. In other words, if student "Brown" is listed in the result of the query, then Brown is in all the classes that Jones is enrolled in, and all the classes that Smith is enrolled in.

Give a tuple calculus expression for part (c).

(c) A list of the names of all students who are enrolled in CSE 5260 or MTH 5100, but not both.

3) (20 points) Let α, β and γ be sets of attributes. The *union rule* states that if $\alpha \rightarrow \beta$ and $\alpha \rightarrow \gamma$, then $\alpha \rightarrow \beta\gamma$.

(a) Prove the union rule using Armstrong's axioms.

Now suppose that δ is also a set of attributes. Then the *double union rule* states that if $\alpha \rightarrow \beta$, $\alpha \rightarrow \gamma$ and $\alpha \rightarrow \delta$, then $\alpha \rightarrow \beta\gamma\delta$.

(b) Prove the double union rule using Armstrong's axioms.

4) Consider the following set F of functional dependencies for the relational scheme R=(A,B,C,D,E,F,G).

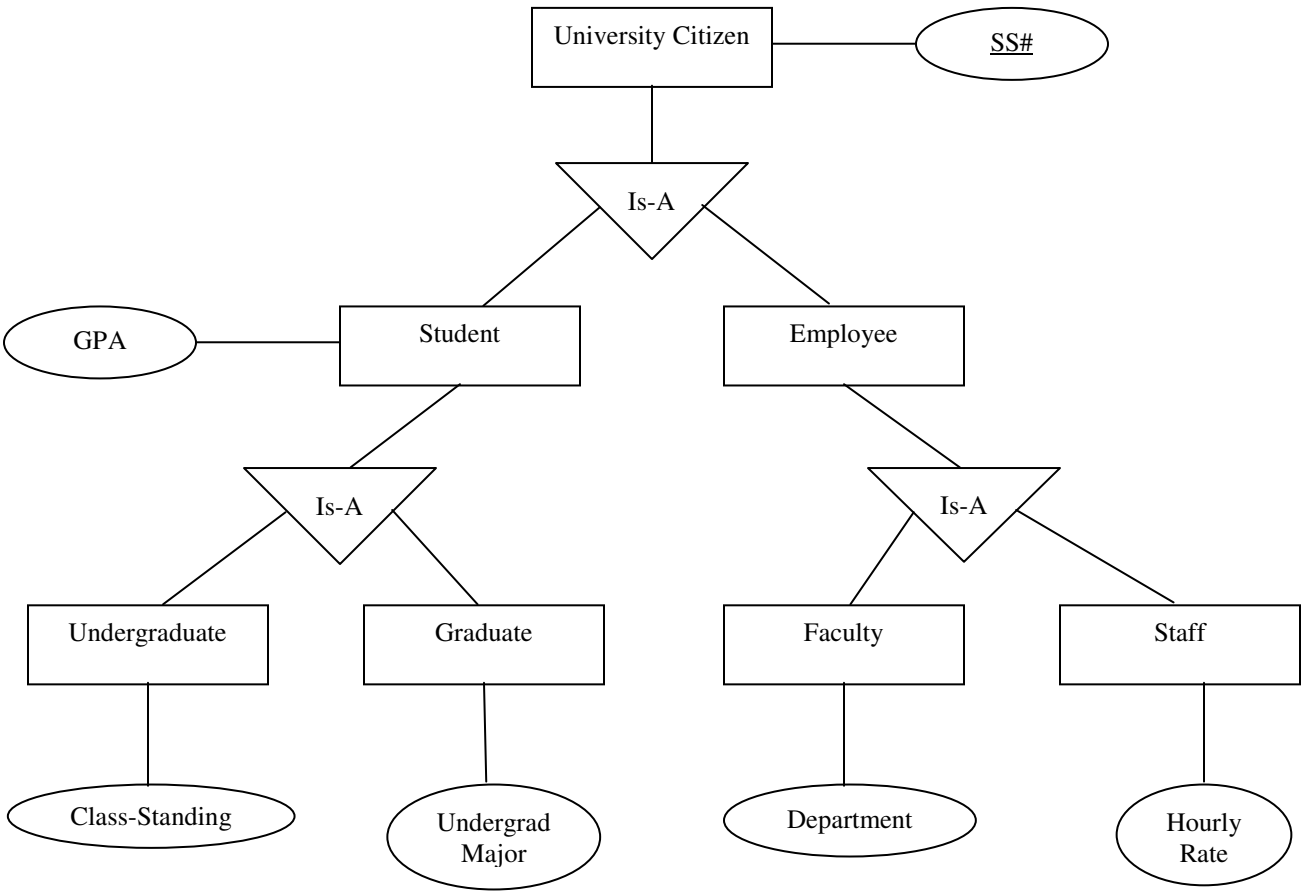
A=>BD
C=>ABE
F=>G

(a) (5 points) Circle each of the following that is a super-key for the above relational scheme.

A C CF AF ABCDEFG

(b) (10 points) Give a decomposition of the above relational scheme into a collection of relational schemes that are in 3NF, have a lossless join and preserves dependencies.

5) (20 points) Consider the following ER diagram.



Give a collection of one or more relational schemes, i.e., tables, for the above ER diagram. Note that your answer should specify all attributes, types, primary keys and foreign keys.