CS 186

MIDTERM

FALL 1994

There are 9 questions to this exam. All count equally.

Queries expressed in relational algebra and relational calculus must follow the syntax used in class. Queries expreseed in SQL must follow the syntax used in class or in the INGRES Manual. All SQL queries should contain NO duplicates in their outputs. It is not necessary to sort the outputs in any order.
You are not allowed to use views for any questions expect question 7.

Use the following relational database schema for all the questions except question 1:
DEPT (dname, location)
STUDENT (name, regno, gpa, level, dept)
COURSE (cno, cname, dept)
TAKE (regno, cno)
None of the columns can have null values.

Problem #1

Suppose there are two relations A (a1, a2, a3) and B (b1, b2, b3). A has two candidate keys and its primary key is (a1, a2). B has only one candidate key and its primary key is (b2, b3). How many candidate keys will the relation A times B have? How many primary keys will the relation A times B have? Please state your reasoning in support of your conclusion.

Problem #2

Express in relational algebra the names of the EECS students who are taking any EECS courses.

Problem #3

Express in relational calculus the names of the departments which either have no students or have at least one student taking each EECS course.

Problem #4

Express in SQL the names and department names of the students who are from departments not located in Evans Hall.

Problem #5

Express in SQL the names of the students who are not taking any EECS courses.

Problem #6

Express in SQL the names of the departments which have the largest number of students taking EECS courses.

Problem #7

Suppose we have the following view definition in the database for all CS students:

```
create view cs_student
as
select s.name, s.regno, s.gpa, s.level
from STUDENT s
where s.dept = 'CS'
```

Suppose we have the following INGRES table level integrity constraints:
create integrity on STUDENT is gpa > 0

Usually the DBMS query modifier converts a query involving views into an equivalent query on the underlying base tables. The query is then modified according to the table level integrity constraints existing on the base tables.

Modify the following query according to the view definition and table level integrity constraint defined above:

```
update cs_student
set gpa = gpa - 1.0
where level = 4
```

Problem #8

Write the rules (and the associated stored procedures) to enforce the integrity constraint: No students can take more than 3 courses from the same department. You may assume that the department of a course will nver change.

Problem #9

Please point out the mistakes in the following ESQL/C program. List the mistakes clearly one by one.

```
exec sql include sqlca;
exec sql include sqlda;
IISQLDA *sqlda;

main()
{
int regno;

regno = 305;

exec sql update STUDENT
set gpa = :gpa
where regno = :regno;
}
```