CS145 Midterm Examination
Thursday, May 8, 1997, 9:30–10:45AM

Directions

The exam is *open book*; any written materials may be used. Answer all four questions on the exam paper itself. The total number of points is 100. Do not forget to **sign the pledge** below.

I acknowledge and accept the honor code.

____

Print your name here: ____

1	2	3	4

Problem 1: (20 points) We wish to design a database consistent with the following facts. Trains are either local trains or express trains, but never both. A train has a unique number and an engineer. Stations are either express stops or local stops, but never both. A station has a name (assumed unique) and an address. All local trains stop at all stations. Express trains stop only at express stations. For each train and each station the train stops at, there is a time.

a) In the space below, draw an E/R diagram representing the information above. Try to make your design follow the design principles of Section 2.3. Indicate keys by underline and many-one or one-one relationships by arrows.

b) Convert your design from (a) into relations, using the E/R-to-relations method outlined in class and the text.

Problem 2: (20 points) We wish to design a database representing employees at a company. Information about the employee includes their unique employee ID, their social security number (also unique), their name, the unique employee who is their manager,† their phone (employees may share phones; no employee has two or more phones), and the set of projects that they work on (projects are identified by unique names).

a) Give an ODL design for the information above, remembering the design principles of Section 2.3. Indicate attribute(s) that are keys. For speed, you may use the following abbreviations: "attribute"→"att", "relationship"→"rel", and "inverse"→"inv".

b) Using the ODL-to-relations strategy outlined in class and in the text, convert your design from (a) to relations.

c) Give a minimal basis for the functional dependencies that hold in your relation(s) from part (b). That is, give a sufficient number of dependencies that any other functional dependency that holds for your relation(s) can be deduced from those you give.

† Don't worry about who the president's manager is. Also, it is not necessary to create a "manager" subclass, although one could do so if one wished.

Problem 3: (30 points) Consider the relation schema $R(A, B, C, D, E)$ with functional dependencies $AB \rightarrow C$, $C \rightarrow D$, $C \rightarrow E$, $D \rightarrow A$, and $E \rightarrow B$.

a) Find all the (minimal) keys for R.

b) Which of the given dependencies violate BCNF?

c) Which of the given dependencies violate 3NF?

d) Using one of your violating dependencies from (b), decompose R into two relations:

e) Give for each relation of your decomposition from (d) a minimal basis for the dependencies that hold in this relation.

Problem 4: (30 points) Suppose we have a database schema

```
Babies(bc#, fn, ln, dob)
Rooms(bc#, date, roomNo)
```

The four attributes of `Babies` represent "birth certificate number" (assumed unique), "first name," "last name," and "date of birth," respectively. A tuple (c, f, l, d) of this relation means that the baby with birth certificate c, first name f, and last name l, was born on day d. A tuple (c, d, r) of relation `Rooms` represents the fact that the baby with birth certificate c occupied hospital room r (assumed unique) on date d (not necessarily its date of birth). Write the following SQL queries and commands. If you know the SQL representation for dates, use it. Otherwise, any reasonable representation will be accepted, e.g., character strings. *Full credit will be given for correct, succinct solutions.*

a) List, in decreasing order, the birth certificate numbers of the babies born on 5/8/97.

b) Find the room numbers occupied by babies with last name "Aiken."

c) Find the "twins," that is, pairs consisting of the first names of babies born with the same last name on the same day. You should list each pair in only one order.

d) Find all the babies without room assignments. That is, produce the birth certificates of the babies mentioned in `Babies` but not in `Rooms`.

e) Find the day on which the largest number of babies were born and the number of babies born on that day.

f) Delete from `Rooms` any tuples that say a baby occupied a room before it was born. (You may assume comparison of dates by $<$ is legal.)