CS482/CS502 – Midterm 2

8:55pm – 10:10 pm

November 16, 2006

Note:

- There are 5 questions (100 points for 25% of the total grade).

- Be short and precise in your answers.

- There are some questions that are different for **CS482** and **CS502**. Be sure that you do the correct one.

Name:

Register for: CS482 or CS502 (Please circle one)

Signature:

1. (15 points) For each class that he is teaching, Professor *Old* keeps a list of students and their score in a table

$$Score(StudID, Score)$$

where $StudID$ is unique. For example, one instance of the list for his database class looks like the following table:

StudID	Score
1	100
2	85
3	70

Normally, Professor *Old* assigns the letter grade using the following criteria: A for $Score \geq 90$; B for $90 > Score \geq 80$; F for everything else. For instance, the above list will become

StudID	Grade
1	A
2	B
3	F

The professor learns about Oracle and he thinks that he can write SQL statements to assign the grade for him. Will you be able to help him by providing him with a set of SQL statement (or a single statement) to create the list with grade letter? (In other words, given the first table, your SQL statement should display the second table as the result).

Answer:

```
SELECT S.STUDID, 'A' AS GRADE
FROM SCORE S
WHERE S.SCORE >= 90
UNION
SELECT S.STUDID, 'B' AS GRADE
FROM SCORE S
WHERE S.SCORE < 90 AND S.SCORE >= 80
UNION
SELECT S.STUDID, 'F' AS GRADE
FROM SCORE S
WHERE S.SCORE < 80
```

2. (25 points) Professor *Old* likes your help so much that now he puts all of his classes in the table with the schema

$$Teaching(CrsCode, StudID, Score)$$

in which he makes sure that the combination of $CrsCode$ and $StudID$ is unique. Furthermore, he also has a table storing the name and address of each student

$$Student(StuID, Name, Address).$$

(a) (15 points of 25 points) Describe the result of the following SQL query in English.

```
SELECT      T.StudID, Avg(T.Score)
FROM        Teaching T
WHERE       CrsCode = 'CS482' or CrsCode = 'CS570' or CrsCode = 'CS220'
GROUP BY    T.StudID
HAVING      Avg(T.Score) >= 90
ORDER BY    T.StudID
```

Sorted list of all students with their respective average in the classes 'CS482', 'CS570', and 'CS220' if the average is greater than or equal 90.

Could also be: Sorted list of all students with their respective average in the classes 'CS482', 'CS570', and 'CS220' if they get an 'A' average.

(b) (10 points of 25 points) It is the end of the year and Professor *Old* would like to congratulate the students who always get A ($Score \geq 90$) in every class that he teaches. Can you help him by getting him the list of students (with name and address) who meet his requirement?

There are many ways to answer the question. You can do (a) get all students with A; (b) get all studentes with something different than A; (c) take the set difference between (a) and (b). In SQL:

```
SELECT DISTINCT T.STUDID, S.NAME, S.ADDRESS
FROM Teaching T, Student S
WHERE T.StudID = S.StuID AND T.Score >= 90
AND
(T.StudID NOT IN
    (
       SELECT A.StudID
       FROM   Teaching A
       WHERE  A.Scrore < 90
    )
)
```

Actually, the statement can be simplified in that you can remove the path $T.Score \geq 90$ in the first WHERE clause (do you know why?); in some situations, it can even be further simplified by removing the Teaching table and the join in the main query as in the following statement (do you know why?)

```
SELECT DISTINCT S.STUDID, S.NAME, S.ADDRESS
FROM Student S
WHERE S.StuID NOT IN
    (  SELECT A.StudID FROM Teaching A WHERE  A.Scrore < 90 )
```

3. (20 points) Consider the following table

A	B	C	D
1	10	a	b
1	9	a	b
2	10	a	d
2	8	a	b
3	7	c	a

It is said that the above table is a valid instance of a relation **R** with the set of attributes $ABCD$ and an unknown set of functional dependencies. Answer the following question with either YES, NO, MAYBE. Provide your justification. (Each question gets 5 points for CS482 and 4 points for CS502 students).

(a) AB is a key of **R**.

Answer: MAYBE. In the current instance AB uniquely determines each row. However, there is nothing that forbids one to add (1, 10, a, c) to the table which will make AB not a key. As such, we do not have enough information to decide whether AB is a key or not.

(b) $CD \rightarrow A$ is a functional dependency in **R**.

Answer: NO. Since the instance contains (1, 10, a, b) and (2, 8, a, b) which indicate that $CD \rightarrow A$ cannot be a FD in **R**.

(c) $ABCD$ is a superkey of **R**.

Answer: YES. Since no instance of **R** can contain two identical rows, $ABCD$ is a superkey.

(d) If $AB \rightarrow C$ is a functional dependency in **R** then $ABD \rightarrow C$ is also a functional dependency in **R**.

Answer: YES. If $AB \rightarrow C$ is a functional dependency in **R** then $ABD \rightarrow CD$ is also a functional dependency in **R** (by augmentation); then, by splitting, we get that $ABD \rightarrow C$ is also a functional dependency in **R**.

(e) **Graduate Students (CS502) only:** A person adds to the above table one more row. After that, he says that $B \rightarrow C$ is NOT a functional dependency in **R**. Is he correct? Why?

Answer: YES. He is correct. Nothing prevents him from adding a row (1, 10, d, c) to the table which will make $B \rightarrow C$ not a FD.

4. (30 points) Consider the relation schema

$$\mathbf{R} = (\{A, B, C, D\}, \mathcal{F})$$

with $\mathcal{F} = \{D \to A, A \to B, B \to D\}$.
Answer the following (10 points each question). Provide justifications for your answers.

(a) What are the keys of $\mathbf{R}$?

One can compute the 16 closures and comes up with the answer: CD, CB, CA are the keys of $\mathbf{R}$.

One can also answer that because C is not in the right hand side of any FD, C must be part of the key. Furthermore, we have that
$$CD_{\mathcal{F}}^{+} = CA_{\mathcal{F}}^{+} = CB_{\mathcal{F}}^{+} = ABCD$$
and none of $A_{\mathcal{F}}^{+}$, $B_{\mathcal{F}}^{+}$, $C_{\mathcal{F}}^{+}$, and $D_{\mathcal{F}}^{+}$ equals $ABCD$. This implies that only CD, CB, CA are the keys of $\mathbf{R}$.

(b) Is $\mathbf{R}$ in BCNF?

NO. Because for $D \to A$ we have that D is not a superkey of $\mathbf{R}$. This is sufficient to say that $\mathbf{R}$ is not in BCNF.

(c) If your answer to the question (b) is NO, find a lossless BCNF decomposition of $\mathbf{R}$. Provide the necessary steps.

First step, we decompose $\mathbf{R}$ to $\mathbf{R}_1$ and $\mathbf{R}_2$ using $D \to A$ where

$$\mathbf{R}_1 = (DA, \{D \to A\})$$

and

$$\mathbf{R}_2 = (DBC, \{D \to B, B \to D\})$$

The first relation, $\mathbf{R}_1$ is in BCNF since D now is the key of it.

The second relation, $\mathbf{R}_2$ is still not in BCNF since D is not a superkey the key of $\mathbf{R}_2$. Using $D \to B$ to decompose $\mathbf{R}_2$, we get the following two relations:

$$\mathbf{R}_{21} = (DB, \{D \to B\})$$

and

$$\mathbf{R}_{22} = (DC, \{\})$$

Both of these relations are in BCNF. So, a BCNF decomposition of $\mathbf{R}$ is: $\mathbf{R}_1$, $\mathbf{R}_{21}$, $\mathbf{R}_{22}$.

5. (10 points) Consider the relation schema

$$\mathbf{R} = (\{A, B, C, D\}, \mathcal{F})$$

with $\mathcal{F} = \{D \rightarrow A, A \rightarrow B, B \rightarrow D\}$.

It is known that for each functional dependency of the form $X \rightarrow Y$ in the set $\mathcal{F}^+$ where X and Y are two arbitrary attributes of $\mathbf{R}$, the relation $\mathbf{R}$ has a join-dependency of the form

$$XY \bowtie \bar{Z}$$

where

$$\bar{Z} = \{X\} \cup (\{A, B, C, D\} \setminus \{X, Y\}).$$

For example, for $D \rightarrow A \in \mathcal{F}$, we have that $DA \bowtie DBC$ is a join-dependency in $\mathbf{R}$. Answer the following (5 points for each question).

(a) Find another JD in $\mathbf{R}$?

We can use $A \rightarrow B$ to find another JD according to the formula and we get:

$$AB \bowtie ACD$$

(A plays the role of D and B plays the role of A, respectively.)

(b) Based on your computation, determine whether $\mathbf{R}$ is in 4NF or not?

NO. Because a JD of $\mathbf{R}$ is $AB \bowtie ACD$ and the intersection of the two sides (left and right) is A which is not a superkey (according to the answer of question 4).