

# **CSC 343 H Y1 Assignment 3**

## **Due date: Monday, July 28<sup>th</sup>, 2003**

### **Rules**

For this assignment you have to submit one report and one archive.

### **Report**

The report should be put in the course drop box, located next to SF/PT bridge on the SF side (second floor), by Monday July 28th 2003 9:00 pm.

The report should contain a cover page where you clearly identify your last name, first name, student id, login name, section (L5101 or L5201). You must put your report in an envelope (letter format) and repeat the cover page information on it.

### **Archive**

You should submit your archive file by the deadline using the cdf facility.

For the Unix command submit, the name of this assignment is: "Assignment3" (submit -a Assignment3 ...). The name of the archive should be:  
`a3\_<studentid>.tar.gz".

### **Late submission**

The late policy is very simple: 20% for each late day regardless whether or not you partially submitted the assignment. The penalty will be calculated on the whole assignment.

Late assignments should be handed in person to an instructor (not to a TA) and the archive file must be sent via email to your instructor.

### **What happens if I do not follow the rules?**

Penalty! We do not have time to find the name for a report or student id for an archive.

### Part I: Functional Dependencies (50 points)

**Question 1. (20 points)** Consider the relation  $R=(A,B,C,D,E,F,G,H,I)$ , and the functional dependencies:  $AB \twoheadrightarrow C$ ,  $C \twoheadrightarrow D$ ,  $E \twoheadrightarrow F$ ,  $FH \rightarrow I$ ,  $GC \rightarrow A$ ,  $FI \rightarrow H$ ,  $D \rightarrow G$ .

- Find all candidate keys.
- Show that  $R$  is not in 3 NF.
- Give a 3NF decomposition of  $R$ . Does this decomposition verify BCNF?
- Give a BCNF decomposition of  $R$ .

For parts (a) , (c) and (d) you must show and explain the steps followed for obtaining your solution. You won't get marks if you just write the solution.

**Question 2. (15 points)** Given a decomposition of schema  $R$  in Question 1, and the following relations:  $R_1 (A,B,C,D)$ ,  $R_2 (C,D,E,F)$ ,  $R_3 (F,G,H,I)$ . Is this a lossless join decomposition?. Is it a dependency-preserving decomposition? . If it is not, which are the lost dependencies?. For both questions you must show the steps you followed for getting your answer.

**Question 3. (5 points)** Given the relational schema  $R(A, B, C, D, E)$ , and the following valid instances of  $R$ :

A	B	C	D	E
A <sub>1</sub>	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>
A <sub>1</sub>	b <sub>2</sub>	c <sub>2</sub>	d <sub>1</sub>	e <sub>1</sub>
A <sub>1</sub>	b <sub>1</sub>	c <sub>2</sub>	d <sub>1</sub>	e <sub>1</sub>

A	B	C	D	E
a <sub>1</sub>	b <sub>2</sub>	c <sub>1</sub>	d <sub>2</sub>	e <sub>1</sub>
a <sub>2</sub>	b <sub>2</sub>	c <sub>1</sub>	d <sub>2</sub>	e <sub>2</sub>
a <sub>2</sub>	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	e <sub>2</sub>

A	B	C	D	E
a <sub>1</sub>	b <sub>1</sub>	c <sub>2</sub>	d <sub>2</sub>	e <sub>2</sub>
a <sub>2</sub>	b <sub>1</sub>	c <sub>2</sub>	d <sub>2</sub>	e <sub>2</sub>
a <sub>1</sub>	b <sub>2</sub>	c <sub>1</sub>	d <sub>1</sub>	e <sub>1</sub>

Do the following Functional dependencies hold on  $R$ ?. Explain why.

- $A \rightarrow BC$
- $D \rightarrow E$
- $\emptyset \rightarrow D$
- $BC \rightarrow D$ .

**Question 4 (10 points).** Given a relational schema  $R(A, B, C, D, E, F)$ , and a decomposition of  $R$  into  $(R_1, R_2, R_3)$ , with  $R_1 = AC$ ,  $R_2 = BCD$  y  $R_3 = BEF$ . If we assume that over any instance of  $R$  the following FD's hold:  $F = \{A \rightarrow DE, D \rightarrow C, BC \rightarrow D, B \rightarrow AF\}$ , show an instance  $I$  of  $R$ , and a tuple  $u$  in  $I$  such that :

- $I$  satisfies the FD's in  $F$ .
- $\Pi_{R_i}(I \cup \{u\})$  satisfies the FD's in  $\Pi_{R_i}(F)$ ,  $1 \leq i \leq 3$ .
- $I \cup \{u\}$  does NOT satisfy the FD's in  $F$ .

## **Part II Embedded SQL (50 points)**

Create an Embedded SQL interface to this database that provides the following query capabilities. Your program should be able to catch DB2 error messages and should give the appropriate message to the user. At the end of each operation the user should get a message indicating whether the corresponding operation was successful. Your program should indicate clearly to the user what data is expected as input and check if the input entered by the user conforms to the requirements. It should not crash, even if the user enters incorrect data.

You will submit an archive file containing every files that you wrote and a Makefile. We must be able to compile your program just typing the following instruction: "a3". In order to provide more freedom the name of the database (c343hxx) will be set via the environment variable DB2DATABASE. Your makefile must use this variable and not hardcoding the name of the database.

**For this program you will use the baseball database used for the previous assignment.**

The program will allow the user to run the following five queries:

**Query 1** ( 8 points): all available information regarding a user-specified playerId.

**Query 2** ( 8 points): the number of Wins and Losses per year for a user-specified playerId.

**Query 3** ( 10 points): insert a new tuple in the Salaries table asking to the user every information that your program might need.

**Query 4** ( 10 points): set a 10% salary increase for all the players of a franchise specified by the user.

**Query 5** ( 14 points): the salary improvement rate per year with the name of the franchise for a user-specified playerId.

In order to compile and link your program for this assignment you have to set the environmental variable LD\_LIBRARY\_PATH

```
setenv LD_LIBRARY_PATH /usr/IBMDB2/V7.1/lib
```