



**CMPS 277-Database System Concepts**

**Fall 2004 – 2005, Final (February 02, 2005 @ 08:00am)**

**Duration: 120 min**

Prof. Leila Ismail and Wadi Jureidini



Student name -----  
 Student ID -----  
 Signature -----

Section      12 MW                  11 TT                  12:30TT

<i>Questions</i>	<i>Points</i>	<i>Grade</i>
1. General (7 questions)	20	
2. E/R Model	10	
3. Relational Model and SQL (3 problems)	20	
4. Integrity and Security (2 problems)	8	
5. Relational Database Design: functional dependencies and forms (4 problems)	12	
6. Object-Oriented and Object-Relational databases	4	
7. Storage and File Structure (2 problems)	8	
8. Indexing and Hashing	22	
Total	104	

Please read the instructions on the next page carefully before you start with the exam.

## IMPORTANT INSTRUCTIONS

1. You **MUST** fill in your name, student ID, and your signature on the first page, and your initials on all the pages, including the first one.
2. You **MUST** explain your answers clearly to get points. Your handwriting **MUST** also be readable to be graded.
3. You **MUST NOT** look at your neighbor's sheet. This will be considered as a cheating attempt.
4. You **MUST NOT** talk during the exam. Any communication will be considered as a cheating attempt.
5. Any cheating will result in a zero on the test and possibly in failing the course and may also lead to disciplinary actions against you.
6. Use your time carefully. If you feel you are stuck, skip to the next question.
7. Use the back side of the pages if the front side is not enough.

## 1. General (20 Points)

There are 7 questions in this section.

- 1) (3 Points) What are the advantages of using a database system instead of using a file processing system? When is it more appropriate to use file processing system rather than a database system?

- 2) (3 Points) What is the difference between procedural data manipulation languages and declarative(non-procedural) data manipulation languages? Compare between the two approaches and give an example of each.

Procedural Languages:

Example:

Declarative (non-procedural) Languages:

Example:

Comparison:

3) (3 Points) What are the main two indicators of a bad database design? What problems do these indicators lead to when found in a database design.

4) (3 Points) What are the different approaches to create an object-oriented database management system? What are the advantages and the disadvantages of each?

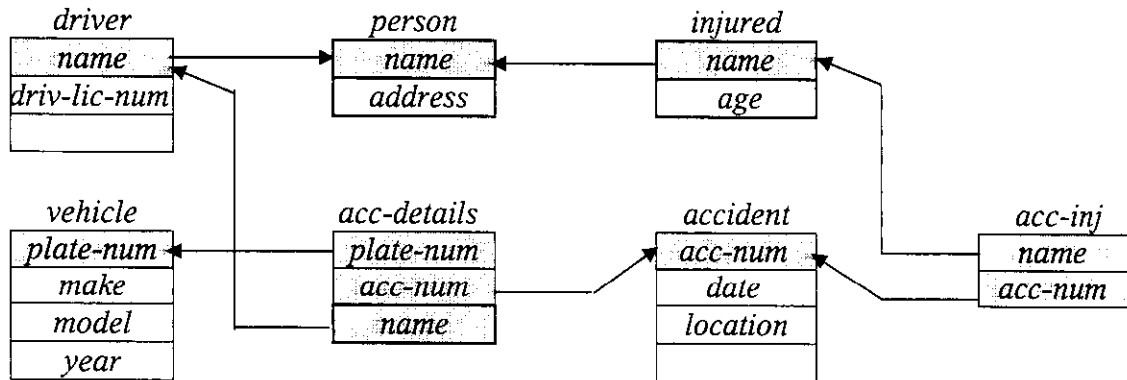
5) (3 Points) What is a view? Why is a view important in database systems?

6) (2 Points) Suppose we examine an instance of a relation  $r(A, B, C)$  and observe that there are no repeated A values among the tuples. Can we assume that A is a key for the relation? Why?

7) (3 Points) "The Buffer Manager in a database management system can be smarter than a buffer manager in an operating system." Comment on this statement, giving an illustration of why it is true.

## 2.E/R Diagram (10 Points)

Considering the increasing number of accidents in Beirut, the Security Authority wishes to automate its accident recording section. Abu Abed, has been hired to design the system, and has come up with the following relational database design:



**Relational design diagram**

The database keeps track of the accidents, the vehicles involved, the drivers, and the injured persons. Each vehicle is identified by its *plate-num* (the registration/plate number of the vehicle), and has a *make* (e.g. BMW), a *model* (e.g. 325I), and a *year*. A person is identified by his/her name, and his/her address is kept in the database. If the person is a driver, then his/her driver's license is stored. If a person gets involved in an accident and is injured, his/her age is stored. An accident is identified by an accident number (*acc-num*), and its location and date are saved.

The details of an accident are to be stored: for every accident we store the vehicles involved, and their drivers.

Give an appropriate ER diagram for the accident recording system, indicating all appropriate constraints.

### 3. Relational Model and SQL

#### Problem 1 (11 Points)

Consider the following relational schema for a the Accident Recording database. (Repeated for convenience)

*driver* (*name*, *driv-lic-num*)  
*person* (*name*, *address*)  
*injured* (*name*, *age*)  
*vehicle* (*plate-num*, *make*, *model*, *year*)  
*accident* (*acc-num*, *date*, *location*)  
*acc-details* (*plate-num*, *acc-num*, *name*)  
*acc-inj* (*name*, *acc-num*)

- a) (3 Points) Express in **SQL**, the following query: "Find the names of drivers who have had more than 3 accidents during the period 01-JAN-2003 till 31-DEC-2004."
- b) (4 Points) Express in **SQL**, and **domain** relational calculus the following query: "Find the name of the injured persons by accidents involving the driver "Khalil Kanaan."
- c) (4 Points) Express in **SQL** the following query: "Find the vehicles with the highest number of accidents."

## Problem 2 (3 Points)

Show how to express  $r \bowtie s$  (left outer join of  $r$  and  $s$ ) using the basic relational operations, plus natural join.

## Problem 3 (6 Points)

Given relations  $r(A, B)$  and  $s(A, C)$ :

- a) (3 Points) Give an expression in SQL that is equivalent to  $\sigma_{B \neq \text{sum}(C)} \Pi_{B,C}(r \times s)$ .

Don't forget that  $\Pi$  eliminates duplicates.

- b) (3 Points) Give a relational algebraic expression equivalent to:

```
select * from r
where exists
(select * from s where s.A = r.A)
```



## 4. Integrity and Security

### Problem1 (3 Points)

Consider the Accident Recording Relational database.

*driver* (*name*, *driv-lic-num*)

*person* (*name*, *address*)

*injured* (*name*, *age*)

*vehicle* (*plate-num*, *make*, *model*, *year*)

*accident* (*acc-num*, *date*, *location*)

*acc-details*(*plate-num* , *acc-num*, *name* )

*acc-inj*(*name*, *acc-num*)

Explain what integrity problems might happen if we insert an *acc-details* tuple. Explain.

## Problem2 (5 Points)

Consider the Accident Recording Database. Abu Abed is the DBA and he wants to have 3 kinds of users, each with different privileges. The three are: the DBA, the Security Authority, and the Insurance Specialist. The Security Authority users have the rights to insert, update and delete data, but not change the database schema. The Insurance Specialists users have the right to read the data without doing any changes.

a) (3 Points) Write SQL authorization statement(s) to enforce these rules on the relation *accident*.

b) (2 Points) Write SQL statements to give the user Adam the same privileges as the Security Authority users.

## 5. Relational Database Design and Forms

### Problem 1 (4 Points)

What, from a normalization perspective, is wrong with a system in which a student-id number encodes year, major and degree into the student-id (e.g. 2004CMPSBS034 means the year of admissions is 2004, the major is computer science, the degree is BS, and 034 is a serial number). Give an example of potential problems it might cause.

### Problem 2 (3 Points)

Consider the following functional dependencies for relation schema  $R = (A, B, C, D, E)$ :

$A \rightarrow BC$ ,  $CD \rightarrow E$ ,  $B \rightarrow D$ ,  $E \rightarrow A$ . Compute  $A^+$

### **Problem 3 (5 Points)**

Give a dependency preserving BCNF decompositions of  $R(N, S, G, L, K)$  under the dependencies  $SG \rightarrow L$  and  $N \rightarrow SG$ .

### **6. Object-Oriented and Object Relational Databases (4 Points)**

Given the nested relation  $r(\text{title: varchar}(20), \text{authors: setof}(\text{varchar}(20)))$ , give an extended SQL query to output a flat table, where each (book, author) pair is separately listed, for example converting  $(\text{DBCConcepts}, \{\text{Avi}, \text{Hank}, \text{Sudarshan}\})$  to  $(\text{DBCConcepts}, \text{Avi})$ ,  $(\text{DBCConcepts}, \text{Hank})$ ,  $(\text{DBCConcepts}, \text{Sudarshan})$ .

## **7.Storage and File Structure**

### **Problem 1 (4 Points)**

What are the main objectives of RAID in a computer system? In what way do the different levels of RAID vary ?

### **Problem 2 (4 Points)**

What is meant by a clustering file organization? When is it useful ?

## 8. Indexing and Hashing (20 Points)

Assume that a B+-tree index is maintained on the relation *accident* of the Accident Recording Database above.

- a) (4 Points) Consider the following instance of the *accident* relation from the Accident Recording Database. Construct a B+-tree with *date* as a search-key. Assume that these values are inserted in ascending order. Also assume that the number of pointers that will fit in one node is 4 pointers. For convenience, abbreviate by x the date x-JAN-2004.

acc-num	date	location
201	01-JAN-2004	Hamra
202	01-JAN-2004	Bashoura
203	02-JAN-2004	Mazra'a
204	02-JAN-2004	Manara
205	03-JAN-2004	Sassin
206	05-JAN-2004	Bliss
207	07-JAN-2004	Hamra
208	07-JAN-2004	Abdel Aziz
209	11-JAN-2004	Hamra
210	17-JAN-2004	Verdun
211	17-JAN-2004	Mazra'a
212	20-JAN-2004	Sadat
213	23-JAN-2004	Manara
214	23-JAN-2004	Hamra
215	29-JAN-2004	Barbir
216	29-JAN-2004	Makdisi
217	29-JAN-2004	Hamra
218	31-JAN-2004	Verdun
219	31-JAN-2004	Mazraa
220	31-JAN-2004	Fassouh

b) (2 Points) Using the B<sup>+</sup>-tree constructed in a), to what does the pointer in a leaf, associated with the value 29-JAN-2004, point to?

c) (2 Points) Use SQL to create an index *acc-index* on the relation *accident* using *date* as a search key.

d) (4 Points) Consider the following SQL statement:

```
Select count(acc-num)
from accident
where d1 < date and d2 > date
```

where d1 and d2 are date values supplied at runtime.

Write pseudo code that represents the execution of the above query using the B+-tree index structure.

e) (2 Points) Illustrate the execution of <sup>d</sup>e) using the instance in a). When  $d1 = 03\text{-JAN-2004}$   
 $d2 = 29\text{-JAN-2004}$