The University of Nottingham
SCHOOL OF COMPUTER SCIENCE
A LEVEL 1 MODULE, SPRING SEMESTER 2007-2008
DATABASE SYSTEMS
Time allowed TWO hours

Candidates must NOT start writing their answers until told to do so

Answer THREE out of FIVE questions

Marks available for sections of questions are shown in brackets in the right-hand margin.

No calculators are permitted in this examination.

Dictionaries are not allowed with one exception. Those whose first language is not English may use a dictionary to translate between that language and English provided that neither language is the subject of this examination.

No electronic devices capable of storing and retrieving text, including electronic dictionaries, may be used.

DO NOT turn examination paper over until instructed to do so
Question 1

(a) Given the relations \textit{Student}, \textit{Marks} and \textit{Module} below:

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
\textbf{Student} & \textbf{ID} & \textbf{Name} \\
\hline
111 & Tom & \\
222 & John & \\
333 & Sue & \\
444 & Anne & \\
\hline
\end{tabular}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
\textbf{Marks} & \textbf{ID} & \textbf{Code} & \textbf{Mark} \\
\hline
111 & G51PRG & 60 \\
111 & G51FUN & 65 \\
222 & G51PRG & 70 \\
222 & G51DBS & 80 \\
333 & G51IRB & 50 \\
333 & G51PRG & 50 \\
\hline
\end{tabular}
\end{table}

\begin{table}[h]
\centering
\begin{tabular}{|c|c|c|}
\hline
\textbf{Module} & \textbf{Code} & \textbf{Title} \\
\hline
G51PRG & Java & \\
G51FUN & Haskell & \\
G51DBS & Databases & \\
G51IRB & Robotics & \\
\hline
\end{tabular}
\end{table}

give the results of the following relational algebra expressions:

\begin{enumerate}
\item \( \pi_{\text{Title}}(\text{Module}) \) \hspace{1cm} (1)
\item \( \sigma_{\text{Code}=\text{G51PRG}}(\text{Marks}) \) \hspace{1cm} (1)
\item \( \pi_{\text{ID}}(\text{Student}) \times \pi_{\text{Code}=\text{G51PRG}}(\text{Marks}) \) \hspace{1cm} (1)
\item \( \pi_{\text{ID}}(\text{Student}) - \pi_{\text{ID}}(\sigma_{\text{Code}=\text{G51PRG}}(\text{Marks})) \) \hspace{1cm} (2)
\item \( \pi_{\text{ID}}(\sigma_{\text{Code}=\text{G51PRG}}(\text{Marks})) \cap \pi_{\text{ID}}(\sigma_{\text{Code}=\text{G51FUN}}(\text{Marks})) \) \hspace{1cm} (2)
\item \( \sigma_{\text{Name}=\text{John}}(\text{Student}) \times \sigma_{\text{Code}=\text{G51DBS}}(\text{Marks}) \) \hspace{1cm} (3)
\end{enumerate}

(b) Assume that there are more values in the tables than shown. In particular, in the \textit{Student} table there might be a student with the name ‘James’ and in the \textit{Module} table there might be a module with the title ‘Computer Graphics’. Write a relational algebra expression which computes James’s mark for Computer Graphics. \hspace{1cm} (5)

(c) Translate the following SQL query into relational algebra: \hspace{1cm} (5)

\begin{verbatim}
SELECT Name 
FROM Student, Marks 
WHERE Student.ID = Marks.ID AND 
Marks.Mark > 50
\end{verbatim}

(d) Translate the following SQL query into relational algebra (using only \( \pi, \sigma \) and \( \times \) operators): \hspace{1cm} (5)

\begin{verbatim}
SELECT Name, Title, Mark 
FROM Student NATURAL JOIN Marks NATURAL JOIN Module
\end{verbatim}
Question 2
You are asked to design a database for an on-line store. The database should contain the following data. First of all, there is data about products that the store sells. Each product has a catalogue number, description, price and amount in stock. There is also customer data: each customer has a name, email address, and postal address (you can also choose to create customer numbers for them). Finally, there are orders which the customers make to buy products. Each order will involve one customer and one or more products. Orders have unique numbers, and they also have status (processing, despatched, delivered).

(a) Draw an entity-relationship diagram for the store database. (10)

(b) Write SQL statements to create the tables (don’t forget to specify primary and foreign keys). (10)

(c) Write SQL statements to insert the following data about the products, using a sequence to generate catalogue numbers starting from 1000 and incrementing by 1: (5)

- Description: suitcase, price: 59.99, amount in stock: 100
- Description: backpack, price: 12.00, amount in stock: 100
Question 3

This question refers to the following tables: Film, Actor, Part, DVD. The primary key for Film is (Title, Year), for Actor the primary key is (First, Last) which stand for the actor's first and last names, for Part it is all the attributes. (Title, Year) in Part is a foreign key to Film and (First, Last) is a foreign key to Actor. For DVD, the primary key is (Title, Year) which is also a foreign key into Film.

<table>
<thead>
<tr>
<th>Film</th>
<th>Actor</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Year</td>
</tr>
<tr>
<td>First</td>
<td>Last</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Part</th>
<th>DVD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Title</td>
<td>Year</td>
</tr>
<tr>
<td>Title</td>
<td>Year</td>
</tr>
</tbody>
</table>

Write SQL queries to do the following:

(a) Find a list of all film directors. (1)

(b) Find a list of first and last names of all male actors (the value of Gender is 'M'). (2)

(c) Find a list first and last names of female actors (the value of Gender is 'F') of the film with the title Magnolia released in 1999. (4)

(d) Return average price of DVDs for each genre of film. (4)

(e) Find all first and last names of actors who acted in any film directed by Anderson. (5)

(f) Update the values in DVD table by adding 1 pound to each DVD price (assume that the price is given in pounds). (4)

(g) Add a new tuple to the DVD table: the film title is Memento, the year is 2000, and the price is 10 (pounds). Assume that the corresponding film is already in the Film table. (2)

(h) What will happen if you attempt to insert the tuple from the previous question when Memento is not in the Film table, and why? (3)
Question 4

(a) Define functional dependency. (2)

(b) Define 2 NF. (2)

(c) Define 3 NF. (2)

(d) Define BCNF. (2)

(e) What are insertion, deletion and update anomalies? (3)

(f) Consider a relation Book with attributes Author, Title, Publisher, City, Country, Year, ISBN. There are two candidate keys: ISBN and (Author, Title, Publisher, Year). City is the place where the book is published, and there are functional dependencies Publisher → City and City → Country. Is this relation in 2NF? Explain your answer. (4)

(g) Is this relation in 3NF? Explain your answer. (5)

(h) Is the relation above in BCNF? If not, decompose it to BCNF and explain why the resulting tables are in BCNF. (5)
Question 5

(a) What is a transaction? (2)

(b) Explain what atomicity of transactions means. (2)

(c) Describe a lost update problem and give an example. (3)

(d) Describe the two-phase locking protocol. (4)

(e) Does using the two-phase locking protocol solve the problem you described in part (c)? Explain why. (4)

(f) Describe the timestamping protocol. (5)

(g) What are the advantages and disadvantages of timestamping compared to using locks? (5)