THE BRITISH COMPUTER SOCIETY

THE BCS PROFESSIONAL EXAMINATION
Diploma

DATABASE SYSTEMS

26th April 2001 – 2.30 p.m. – 4.30 p.m.
Answer FOUR questions out of SIX. All questions carry equal marks
Time: Two hours.

*The marks given in brackets are **indicative** of the weight given to each part of the question.*

1. *a)*   What features would you expect to find with each of the following database tools:

- Command line interface;
- Form editor;
- Report writer?

Illustrate your answer with examples from database systems you have used.   **(9 marks)**

b)   SQL is not suitable as a language for writing database applications. Why is this?   **(6 marks)**

c)   To overcome the limitations of SQL, database application programmers can resort to one of the following three solutions:

- Embed SQL statements in a general-purpose programming language like C or COBOL;
- Use a call-level interface with a general-purpose programming language;
- Use an extension of SQL that provides the features of a general-purpose programming language.

Describe the features provided by each of these solutions. Illustrate your answer with examples. **(10 marks)**

2. *a)*   What are the main distinctions between the database approach and the file-processing approach to data processing?   **(8 marks)**

b)   Two of the main functions of a database administrator are authorising access to the database and managing the performance of the database system.

i)   Describe the features of SQL used by a database administrator for authorising access to the database.

(8 marks)

ii)   Describe the basic principles managing the performance of the database system.   **(9 marks)**

3. A bank requires a database designed to keep track of the accounts of its customers. Each customer may have a number of accounts. Each of these accounts may be in the name of one or more customers. Each account is located in a particular branch of the bank. In addition, each customer may have zero or more loans with the bank. Each loan must be with one or more customer. Each loan is also located in a particular branch.

The database must record the following information:

- The name of the bank, its address, and its main telephone number;
- The address of each branch, the sort code for each branch;
- The name of each customer, the customer's address and the customer's telephone number;
- For each bank account, the account number, its balance, and the type of account;
- For each loan, the loan account number, the outstanding balance, and the interest rate.

The database must also record the various relationships between the different entities in the database.

a) Draw an entity-relationship diagram for the database. In your diagram, illustrate which attributes are key attributes, which relationships are optional and mandatory, and the cardinality of the relationships.
(9 marks)

b) Define a set of database tables that implement the entity-relationship diagram. Indicate which attributes will represent the primary key for each table, which attributes are foreign keys, and which attributes cannot have a null value. **(9 marks)**

c) The referential integrity rule does not allow unmatched foreign key values. What would be the consequences of this rule for your database design? **(7 marks)**

4. Consider the following two database platforms:

PLATFORM 1:
File server platform: with a file store deployed on a server running network software and many client PCs, each running database packages such as MS-Access.

PLATFORM 2:
A web based client server platform: with a database system deployed on a web server and many client PCs querying and updating a database over the Internet, using a web browser such as Netscape or Internet Explorer.

a) Outline the functional differences between each of the above platforms in terms of processing distributed data. **(8 marks)**

b) Describe an application that would use PLATFORM 2 above. In particular describe a range of querying and updating activities that your application requires. **(8 marks)**

c) Describe a technology that supports the processing of dynamic data content in web pages, based on PLATFORM 2 above. **(9 marks)**

5. Examine the two Tables AUTHORITY and CONFLICTS given below.

Table AUTHORITY shows which user is authorised to run a transaction. For example, Bill is authorised to run transaction Ta.

Table CONFLICTS below contains pairs of transactions that conflict. For example, Tb conflicts with transaction Tc.

AUTHORITY

USER	TRANS
Bill	Ta
Bill	Tb
John	Tc
John	Tb
John	Tf
Alex	Tf

CONFLICTS

T1	T2
Tb	Tc
Tf	Tc
Ta	Tc

a) Using SQL list those Users who have authority to access more than one transaction.   **(8 marks)**

b) Using SQL, construct a View called 'PAIRS' that finds all the pairs of transactions that each User can access.   **(10 marks)**

The View PAIRS would contain the following sample data:

PAIRS

USER	TRANS1	TRANS2
Bill	Ta	Tb
Bill	Tb	Ta
John	Tc	Tb
John	Tb	Tc
John	Tb	Tf
John	Tf	Tb
John	Tf	Tc
John	Tc	Tf

c) Using SQL, create another View that finds which user has access to each conflicting pair of transactions.

For example the View should return:

USER John *has access to the conflicting transaction pairs* Tb, Tc and Tc, Tf   **(7 marks)**

[Turn over

6. *a)* Describe the features of a commercial visual software development tool/environment supporting the generation of screen Forms, used for developing user interfaces to database applications.   **(7 marks)**

    b) Draft out the design for a sequence of screen Forms that are required to meet the specification given below.

    Your design should include:
    -   Diagrams of screen shots showing how the user would interact with your Forms.
    -   The processing requirements of data input and output between the database and the User Interface.
    -   An indication of how your design ensures the integrity of data input by the user.   **(18 marks)**

    Specification:
    A computing department in a college has a client-server computer network in which students progress through four Computer Aided Learning (CAL) software packages running on the client PCs. The college wants to monitor and record student access to the software packages in a session log. Students have access to the CAL software packages during a one-hour period each week, for a total of six weeks. A database Table called 'Logins' stores the session logs and contains the following structure:

    SessionID (the primary key),
    StudentID,
    SoftwarePackage,
    Log-in Time,
    Log-out Time,
    Login Date.

    A student can only access one software package at a time but may access another package during the same one-hour period. There may be many students accessing the same package during a session, therefore the SessionID is unique for each login.

    Another Table called 'Students' contains data on students, indicating who are registered users of the system:

    StudentID (the primary key)
    StudentName
    CourseName

    To log-in to the software package a student supplies their StudentID followed by their CourseName. After successfully logging in the student selects the software package they wish to use and continues until they have finished the session.

    When a student has finished they must log-out and their session log is updated with the log-out time and date. Students can subsequently log back in again within the hour period and start another session or log out altogether.