UNIVERSITY OF LONDON
Goldsmiths College

BSc Examination 2001

COMPUTING AND INFORMATION SYSTEMS

IS52003A (CIS209) - Database Systems

Internal

Duration: 3 hours

This paper is divided into three parts: Part A, B and C. **All parts are compulsory**. Each part consists of problems. Within each part, you are required to attempt **one problem only**.

The marking scheme has been made available for you. Part A carries 40 marks. Part B carries 30 marks. Part C carries 30 marks. Within a part, all problems carry equal marks. The number of marks carried by each question, within a problem, are printed within square brackets, aligned to the right.

Electronic calculators are not necessary for this assignment, therefore they should not be used.

THIS EXAMINATION PAPER MUST NOT BE REMOVED FROM THE EXAMINATION ROOM.

Part A

Attempt **one** of the following two problems
Total marks = 40; both problems carry equal marks.

Problem 1. Total marks: [**40**]

a) Define an ER model/diagram that describes the data to be stored in a database for a sports competitions organiser, according to the requirements specification provided below. Your final answer should consist of a model that is readily translatable into a relational model; however, if you cannot achieve this requirement, provide the best answer you are capable of. [**30**]

b) Transform the ER model into a relational model, specifying the primary, alternate (if any) and foreign keys of each resulting relation. [**10**]

*OPTION: You may disregard the last two paragraphs (in italics) of the requirement specification. If you do so, the maximum marks you can achieve are **25** for a) and **7** for b).*

A company called *Sports-Events* wants to build a database to store information about the competitions they organise. A part of the data they require is described below.

Each competition is described by: a name, a list of regulations (the list may be though as being just one item of data, for no requirement exists to access parts of a certain list of regulations), a deadline, by which all the contenders must register, and an in-house generated id for a convenient identification (unique per competition). Contenders must register for each competition they want to participate in. The date of the registration needs to be recorded. A contender is allowed to register for more than one competition. Each competition has a winner who has to be recorded in the database. A contender is described by: an alias/name (of their choice), an address, a telephone number and gender. Contenders, too, are given an in house generated id for an easier identification.

A competition consists of events. Usually, a competition has more than one event. A competition has no events if it has no registered contenders (which is possible). A certain event belongs to a single competition. An event is described by: a level (e.g. "qualifications", "semi-finals", "final", etc.), a number (e.g. number = 2 and level = "semi-finals" means the second semi-final of the competition), a day (when it takes place), a (starting) time and a venue. Contenders participate in events and for each such participation, they obtain a result which is to be stored in the database. Contenders qualify for the (i.e. will be participating in) events of the next level of a competition based on their results in the events of the previous level (of the same competition).

Events are supported by support-staff. A support-staff is described by: an in-house generated id (unique per staff; constructed from the initials of the staff member), a name, a telephone number, a location/office and a type (e.g. "security", "doctor", "referee", "cleaner", etc.). One support-staff may be assigned to more than one event and one event may require more than one support staff. Each event must have at least one support staff. Some support staff may never be allocated to events. An assignment of a support staff to an event may be compulsory (case in which the respective staff must attend the event) or optional; this information will have to be recorded in the database.

A contender can be either an individual or a team. In the case of an individual, the following extra-information must also be stored in the database: the actual name (which may coincide with the alias/name or may not) the age and a list of known medical problems (the list may be thought as being just one item of data, in that no parts of it need to be separately accessed). In the case of a team, the following extra-information must also be recorded in the database: a medical certificate endorsed by a doctor which states the fitness of the team, a list of team members (this list, too, may be thought of as being just one item of data) and the team captain's name.

Support staff, as already stated, is of different types. If a support staff is "security", then the following extra information must also be stored in the database: a role (e.g. "contender's body guard", "public security" etc.) and gender. If a support staff is a "doctor", then the following extra information must also be stored in the database: speciality and level of intervention (e.g. "consultant", "partial intervention", etc.). If a support staff is a "referee", then the following extra information must also be stored in the database: accreditation (e.g. "local", "regional", "national", etc.).

Problem 2.	Total marks: [**40**]

Consider the relation "Sport-Events" presented below.

Contender	Address	Competition	RegDate	Round	Result	Position	Requirements	Winner

Assume that it already is in first normal form (1NF). The meaning of its attributes is: "Contender" and "Address" represent the name and the address of a contender, respectively; "Competition" represents the name of a certain competition; "RegDate" represents the date when a contender registered for a certain competition; each competition consists of "Round"s (typical values include: 'preliminary qualifications', 'first round; qualifications", "semi-finals", etc.); contenders participate in rounds and for each such participation they obtain a "Result" and a particular "Position" (e.g. for the semi-finals of 400m, M. Johnson had the result '52:12' seconds and was on the 'first' position); each round has one set of requirements (the set is considered atomic) stored in "Requirements", on the basis of which contenders are selected from the previous rounds (typical values, for example for running, may refer to position; for example, for the semi-finals of 400m, the requirements are 'the best 16 from qualifications'); each competition has a "Winner". Further, consider the following assumptions:

- a contender only registers once for a certain competition;
- each contender has a unique name;
- a contender may register for more than one competition;
- only one address is required for each contender;
- a certain competition, usually, has many rounds (e.g. 'qualification', 'semi-final' and 'final');
- a contender has one result and one position per round per competition; however, not all the contenders registered for a certain competition participate in all its rounds;
- each competition has a unique winner.

a) Provide an extension of this relation consisting of (not less than) 5 tuples. [**3**]

b) This relation is susceptible to update anomalies; provide an example of each of an insertion, a deletion and a modification (update) anomaly, based on the extension you illustrated for the previous point. [**7**]

c) Bring "Sport-Events" to Boyce-Codd normal form (BCNF); if you prefer, you may do it directly, i.e. without going through the intermediate forms 2NF and 3NF. Obviously, the justification of each normalisation step must be based on the functional dependencies you have identified from the problem's assumptions; state them explicitly. Specify the primary key, the alternate keys (if any) and the foreign keys for each resulting relation. [**20**]

d) Separately from the above relation, consider the following relation:

SportEvent	Date	Place	OfficialReferee

with the following non-trivial functional dependencies:

SportEvent, Date, Place $\rightarrow$ OfficialReferee

OfficialReferee $\rightarrow$ SportEvent

Discuss the problems associated with the normalisation of this relation to BCNF. [**10**]

Part B

Attempt **one** of the following two problems
Total marks = 30; both problems carry equal marks

Consider the following database schema, defined in SQL.

```
CREATE TABLE Patients (
     Patient_id          SERIAL,       --automatically generated
     Name                VARCHAR(30),
     Address             VARCHAR(50),
     Sex                 CHAR(1),
     Dob                 DATE,
     UNIQUE              (Name, Dob),
     PRIMARY KEY         (Patient_id));
CREATE TABLE Doctors (
     Doctor_id           CHAR(4),      --in house created code
     Name                VARCHAR(20),
     Speciality          VARCHAR(50),
     Max_no_treatments   INT,
     PRIMARY KEY         (Doctor_id));
CREATE TABLE Diagnoses (
     Dia_id              SERIAL,       --automatically generated
                                       --may be used for joins
     Patient             INT,
     Symptoms            VARCHAR(200), --diagnostic description
     Dia_date            DATE,         --when the diagnosis was made
     Doctor              CHAR(4),      --and by whom
     Di_description      VARCHAR(100),
     Class               VARCHAR(50),  --class of the diagnostic
     PRIMARY KEY         (Dia_id),
     UNIQUE              (Patient, Symptoms, Dia_date, Doctor),
     FOREIGN KEY         (Patient) REFERENCES Patients(Patient_id),
     FOREIGN KEY         (Doctor) REFERENCES Doctors(Doctor_id));
CREATE TABLE Treatments (
     Treatment_id        SERIAL,       --automatically generated
     Diagnosis           INT,
     Doctor              CHAR(4),
     Tr_description      VARCHAR(200),
     PRIMARY KEY         (Treatment_id),
     UNIQUE              (Diagnosis, Doctor),
     FOREIGN KEY         (Diagnosis) REFERENCES Diagnoses(Dia_id),
     FOREIGN KEY         (Doctor) REFERENCES Doctors(Doctor_id));
CREATE TABLE DAdmin(--daily administration of drugs
     Treatment           INT,
     Drug                VARCHAR(40),
     Admin_time          TIME,         --administration times
     Qty                 INT,
```

If the schema is not sufficiently clear, you may want to read the following explanations.
This schema describes a part of the database of a hospital. Patients are diagnosed and treated by doctors. The information about patients and doctors is recorded in “Patients” and “Doctors” respectively. Patients are diagnosed by doctors for different symptoms. On a certain date, for a specific set of symptoms a patient is given a diagnostic by a doctor. A certain set of symptoms presented by a user and considered in a diagnostic are stored in the attribute “Symptoms”; therefore, “Symptoms” holds a set of symptoms (e.g. “headache, fever, nausea”). A patient may be diagnosed by more than one doctors, even for the same symptoms and even on the same date. A patient may be diagnosed on different dates and may have different sets of symptoms. The information about diagnostics is stored in “Diagnoses”; “Di_description” represents the description of a certain diagnosis and “Class” represents a standard classification of the diagnosis. Only some of the diagnoses that a patient was diagnosed with are treated. A certain diagnosis might be assigned to more than one doctor. There is no constraint to have the doctor who put a diagnostic as one of the doctors who treat the respective diagnostic. Note that the patient involved in a treatment can be identified only via his/her diagnosis. Some of the treatments involve the administration of drugs; one treatment may require the administration of more than one drug. The information about treatments is recorded in “Treatments”. The daily administration of each drug per treatment – as: drug, time and quantity – is recorded in “DAdmin”.

Problem 1. Total marks: [**30**]

Based on the database schema previously described, express the following natural language queries in SQL.

1. Find the maximum number of treatments that doctor ‘Michael Smith’ would normally be involved in. [**2**]

2. Find whether or not there is a treatment that includes the administration of the drug ‘betamicin’. [**2**]

3. The patient with the id 34459 was diagnosed with ‘lung infection’ on ‘20/05/2001’ by the doctor with the id ‘MSS1. Find the patient’s symptoms on the basis of which this diagnosis was given. [**2**]

4. Find the number of times per day and the daily total quantity the drug ‘penicillin’ is administered for the treatment 17765. [**3**]

5. Find all the patients (name, address, sex and dob) who were diagnosed on 23/12/2001. [**3**]

6. Find the names of all the patients who presented the symptom of a headache after ‘12/01/2001’ (note that the field “symptoms” holds a list of symptoms; e.g. “suffocation and vertigo”, “headache, nausea, fever”, etc.). [**3**]

7. Find the number of treatments each doctor is in charge of; the result should contain the name of each doctor and his/her number of treatments. [**3**]

8. Find all the doctors who treat more than 5 different patients (if a doctor is in charge of more than one treatment for a certain patient, then the respective patient should only be counted once); the result should contain the name of each doctor and his/her number of patients. [**4**]

9. Find the name and the address of all the patients who are treated by doctor ‘Michael Smith’. [**4**]

10. Find all the patients – id, name, address, sex, dob – who were diagnosed by doctor ‘Michael Smith’ but are not treated by him. [**4**]

Problem 2. Total marks: [**30**]

Based on the database schema previously described, express in SQL the following natural language data constraints and security rules (if you need to modify the table definitions then just provide the modification and not the whole definition for the table).

1. The date of birth of a patient cannot be after the date when it is inserted (hint: assume that an SQL unary operator "CURRENT_DATE" exists). [**2**]

2. The speciality of each doctor must be inputted in the database; in other words, no doctor record can exist in the database without a speciality. Also, if a value for the maximum number of treatments is not inputted for a certain doctor, this should automatically be assigned to 30. [**2**]

3. If a patient is removed from the database, the information about his/her diagnoses and treatments must automatically be removed. [**3**]

4. A doctor can diagnose only within his/her speciality; this means that the "Class" associated with a diagnosis given by a certain doctor must exactly match the "Speciality" of the respective doctor. [**4**]

5. For no treatment can the daily quantity of administered 'penicillin' exceed 100mg. [**4**]

6. According to the existing schema, different doctors may treat a patient for a certain diagnosis. Now, assume that only one doctor may treat a patient for a certain diagnosis and this doctor should be the same with the one who put the diagnosis. Modify the schema in order to reflect this latter assumption. [**3**]

7. Allow everybody to see the name and speciality of each doctor. [**2**]

8. Allow the patient Joe Hunt, having the username 'pat01jh1', to see the list of all the diagnoses he received, in the form of symptoms, date, doctor, diagnosis. [**3**]

9. Allow all patients to see their own diagnoses, presented in the form of symptom, date, doctor, diagnosis. Assume that the table "Patients" was modified such that it includes a field "User_name" (representing the user name of each patient) (hint: use the SQL function "user()"). [**4**]

10. Allow the patient Joe Hunt, having the username 'pat01jh1', to see the doctor and description of all the treatments he is receiving. [**3**]

Part C

Attempt one of the following two problems
Total marks = 30; all problems carry equal marks

HINT: Gauge the length of your answers by the number of marks awarded.

Problem 1. Total marks: [**30**]

a) Enumerate and provide a brief explanation for (at least) four main features of a database management system. [**6**]

b) Two transactions, A and B, need to retrieve and then to update (in that order) the same data object, r. Diagrammatically illustrate and explain how one of the updates can be lost, due to a concurrent access. Further, diagrammatically illustrate and explain how, by applying the data access protocol to this problem, a deadlock may be generated. Lastly, explain how the deadlock situation is identified and dealt with by a DMBS. [**15**]

c) Define and explain the notion of a serialisable set of transactions. Under which circumstances is any schedule of a set of transactions guaranteed to be serialisable? [**9**]

Problem 2. Total marks: [**30**]

a) Compare, from the point of view of data management, the file based approach with the database approach to the development of software systems (you may want to employ a diagram in your answer). [**5**]

b) Define and explain the term “transaction”. Describe the mechanisms provided by transaction managers (part of DBMSs) whereby transactions are implemented at the application’s end; for this description, utilise an example of your choice, in a pseudo-code of your choice (if not self explanatory, explain the components of the pseudo-code you are using). [**10**]

c) Discuss the issue of database recovery. [**15**]