UNIVERSITY OF LONDON

Goldsmiths College

BSc Examination 2003

COMPUTING AND INFORMATION SYSTEMS

IS52003A (CIS209)

Database Systems

Internal

Duration: 3 hours

This paper consists of **5** questions. Each question carries **25** marks. Answer only **4** of them. You may choose **any 4** questions. Full marks will be awarded for **complete** answers to **4** questions.

The mark carried by each part is printed within square brackets. **Gauge the time to be spent on each part by the number of marks awarded**.

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

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

QUESTION 1
Topics covered: generalities about database systems and the relational model.

Part 1 [6]

A database is defined as "*a shared collection of logically related persistent data as part of the information system of an organisation*". Explain in brief the meaning of "shared", "logically related" and "persistent" in this definition.

Part 2 [6]

Define the notion of "candidate key". Can a relation have more than one candidate key? Give an example. Define the notion of "primary key" in terms of the "candidate key".

Part 3 [3]

Explain in brief what is it meant by program–data independence in the context of database systems.

Part 4 [3]

Enumerate three benefits of the database approach to data management (as opposed to a bare file based approach).

Part 5 [4]

Data can be stored in files and application programs can share this data by having a direct access to the respective files (refer to Diagram 1, below). However, data-centred applications normally employ a database management system (refer to Diagram 2, below).

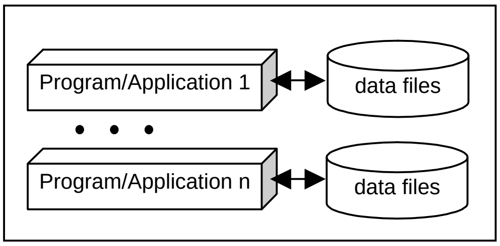

Diagram 1

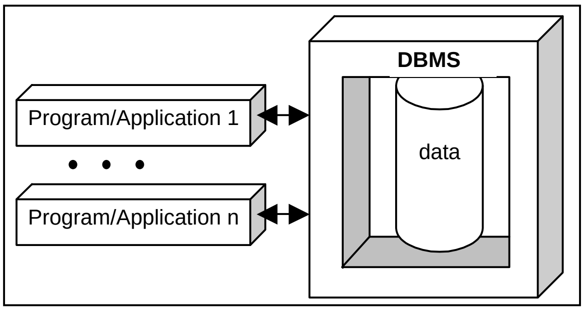

Diagram 2

Discuss the *redundancy of data* in each of the two approaches.

Part 6 [3]

A union, intersection or difference can only be performed between two relations if they are *type compatible*. What is it meant by type compatibility? Give an example of two type compatible and two non type compatible relations (only the headings are required).

QUESTION 2
Topics covered: ER/EER modelling.

Part 1 [19]

Draw an ER diagram for the following description. The diagram should illustrate the entity types, including their attributes, the relationships between them and the multiplicity of each relationship (note that the textual description does not specify the multiplicity of all the relationships; you will have to state it yourself, according to your understanding of the problem). Work according to the following conventions: many-to-many relationships do not have to be transformed into one-to-many relationships; attributes could be composite and/or multi-valued; relationships may have attributes.

> An airline company intends to develop a database system for storing information about flights and their passengers. The information about flights is of two types: *general* flight information (such as the BA123 flight from London to Paris, departing at 12:00, available daily) and *specific* flight information (such as the BA123 flight on 02/04/2003 whose captain is John Smith). Note that a passenger may book a place on a specific flight but could not book a general flight. The general flight information should consist of flight number, destination airport, starting-point airport, intermediate stops (a list of airports), departure time and arrival time. The specific flight information should consist of date of flight, captain, delay at departure and delay at arrival.
> The airline company has a flotilla of aircraft. The required information on aircraft consists of model/type (e.g., Airbus 540), seat capacity, normal flying altitude, flight autonomy (how long could it fly without refuelling), and an internal aircraft identifier used in case the company has more than one aircraft of the same model/type. A specific flight must have assigned one aircraft.
> Passengers book (specific) flights. The information regarding a booking that is to be stored in the database consists of ticket number and payment details. It would be useful to also have recorded in the database the time when the booking was made and the name of the staff member who performed the operation. The information required for each passenger consists of name, contact details — these are made of house number, street, city, postcode, country and telephone — and whether the person is smoking or non-smoking.

Part 2 [6]

Consider the ER structure depicted in Figure 1 below. Find an application (e.g. library, hospital, software development company, sport events, university, etc.) for which this structure could be used to model a part of its information system and illustrate this model — i.e., find meaningful names for the entity types (E_1 and E_2), for the attributes of each entity (a_1, ..., a_5, b_1, ..., b_4), for the relationship R and for its attribute c.

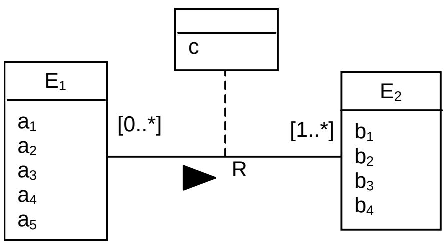

Figure 1

QUESTION 3
Topics covered: functional dependencies, non-loss decomposition, normal forms (up to BCNF) and dependency preservation.

Part 1

Consider the following relation.

patient_id	patient_name	p_DOB	disease	doctor	speciality	diagnosis	treatment

Consider also the following assumptions: a doctor gives a unique diagnosis for the disease of one patient; however, a doctor may give different diagnoses for the same disease (for different patients); each diagnosis has associated a unique treatment; a doctor has a unique speciality; a patient has a unique patient_id.

a) In each of the expressions below, substitute the question marks with sets of attributes from the above relation to obtain expressions representing functional dependencies. [3]

patient_id → ?
? → diagnosis
? → treatment

b) Choose a primary key for this relation. [1]

Part 2

Consider the following relation.

project	task	max_budget	duration	payment_rate	contractor	contracted_time

and the following functional dependencies:

(project, task) → max_budget, duration
//there is a unique max_budget and period of work (duration) per project task
(task, max_budget, duration) → payment_rate
//the contracting payment rate is unique given a certain task, max_budget and duration
(project, task, contractor) → contacted_time
//contractors are employed on project tasks

Assume they completely express all the functional dependencies existing in the given relation (i.e., the other are either trivial or can be deduced from the given ones).

a) State the primary key for this relation (there is a unique candidate key). [2]

b) State the definition for Boyce-Codd Normal Form (BCNF) [2]

c) State a reason why this relation is not in BCNF. [2]

d) State Heath's theorem (for non-loss decomposition). [3]

e) Decompose/transform (non-loss) the given relation into a set of relations in BCNF. Explain how you apply Heath's theorem for each decomposition you make. State the end result clearly. Also, state the candidate keys for each resulting BCNF relation. [12]
Note that the order in which you employ the above functional dependencies in normalisation is important — some orders may lead to the loss of certain dependencies. You are advised to start with the second functional dependency.

QUESTION 4
Topics covered: SQL (data manipulation and integrity constraints).

Consider the following database schema (some tuples are provided for explanatory purposes; arrows denote foreign keys; primary keys are in bold and underlined):

Customer

name	dOB	address	occupation

AccountType

name	category	minBalance	interest
Student Current	current	-500	0.5%
Golden Current	current	-2000	1%
Golden Savings	savings	5000	3.5%

Account

accNo	*type*	*owner*	balance
C-110-221	Student Current	Joe Bloggs	-245
S-009-677	Golden Savings	Mary Bear	5500

Transaction

accNo	date	time	transType	valIn	valOut
C-110-221	12/04/2003	12:30	cash withdraw	0	30
C-110-221	21/04/2003	9:15	cheque payment	200	0

Part 1

Referring to the above schema, express the following natural language queries in SQL:

a) Find the minimum balance and interest for the 'Golden Savings' account type. [1]

b) List the account names and their corresponding interest rates ordered according to the interest rates for all the 'savings' account types. [2]

c) List the date, time, transaction type, value-in and value-out for all the transactions incurred between '1/01/2003' and '1/04/2003' on 'Joe Bloggs''s 'Student Current' account. [3]

d) List how much money 'Mary Bear' has in all her 'savings' accounts. [3]

e) List the name, address, occupation and total balance for all the customers whose total balance is negative — for each customer, their "total balance" means the sum of the balances of all their accounts. [3]

f) List the account number, balance, account name/type and the interest on the respective account, for the account on which 'Joe Bloggs' has the highest balance. [3]

g) List the account category, number of accounts and average balance per category of account for all the customers whose occupation is 'student'. [4]

Part 2

Referring to the above schema, express the following integrity constraints in SQL:

a) The balance on each individual account (stored in 'Account') should not go below the minimum balance for its type (as stated in 'AccountType'). [3]

b) The value-out for cash withdraws (see Transaction) for any "Student Current" account cannot be greater than 100 per individual transaction. [3]

QUESTION 5
Topics covered: views, security and transaction processing.

Part 1

a) Consider the relation "Absences (student, date)", with the primary key "(student, date)", which records the dates when students are absent from university. For illustration, a small extension is given in Figure 1 below:

Absences

student	adate
S. Allen	12/01/2003
P. Clark	03/02/2003
S. Allen	05/03/2003
M. Lewis	05/03/2003

Figure 1

```
CREATE VIEW AbsCount AS
   SELECT   student, COUNT(adate) AS noAbsences
   FROM     Absences
   GROUP BY student;
```

Figure 2

Consider the view "AbsCount", as defined in Figure 2. This represents the number of absences per student. Lastly, consider the following update operation attempted on AbsCount:

`UPDATE AbsCount SET noAbsences = 3 WHERE student = 'S. Allen';`

Discuss how this update operation could/should be dealt with by a relational DBMS and draw a general rule regarding views. Although the syntax is that of SQL, you should consider the problem independent from any specific database language and/or DBMS. [6]

b) State two restrictions imposed by SQL2 on update operations to views. [4]

Part 2

a) What is a transaction? Give a simple example. [4]

b) State and succinctly explain the two mechanisms customarily provided by the transaction manager (of a DBMS) for the implementation of transactions. [4]

Part 3

a) Explain in brief what is it meant by impedance mismatch. [4]

b) Consider two real life systems, A and B. Each requires the support of a database system. System A consists of very many *types of data objects* (or *entities*), but each type (or entity) has only a few instances. System B consists of a moderate number of *types of data objects* (or *entities*), but each type (or entity) has very many instances. Disregarding any other constraints, for which system would you propose the use of a relational DBMS? [3]