UNIVERSITY OF LONDON

Goldsmiths College

BSc Examination 2002

COMPUTING AND INFORMATION SYSTEMS

IS52003A (CIS209)

Database Systems

Internal

Duration: 3 hours

This paper consists of **5** problems. Each problem carries **25** marks. Answer only **4** of them. You may choose **any** 4 problems; full marks will be awarded for complete answers to **4** problems. Problems are made of questions. For each problem you choose, attempt **all** its questions.

The mark carried by each question is printed within square brackets. Gauge the length of each answer by the number of marks awarded.

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.

PROBLEM 1 [25]

Topics covered: generalities about databases ; generalities about the relational model.

Question 1
Define the relational model. What is a relational database management system (DBMS)? **[3]**

Question 2
Define the notion of "foreign key". Give an example. **[3]**

Question 3
Explain the two types of program–data independence on the basis of the three level ANSI/SPARC architecture of a database system. **[4]**

Question 4
What is a system catalogue? Give examples of two types of data/information it usually includes and, in each case, explain the (potential) use of this data/information (one or two sentences per type of data/information). **[5]**

Question 5
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).

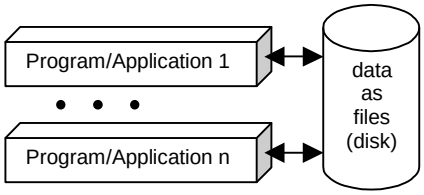


Diagram 1

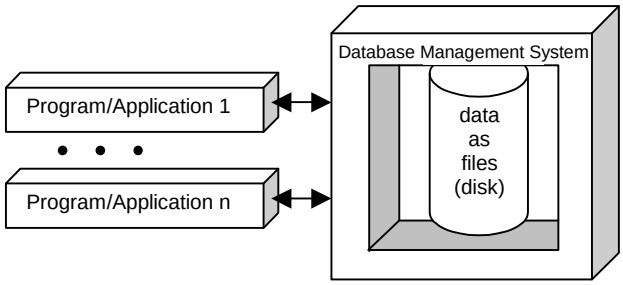


Diagram 2

State why the latter approach (Diagram 2) is preferred for data-centred applications (refer to at least two features of a DBMS). **[5]**

Question 6
Explain what is it meant by impedance mismatch, in the context of relational database systems. **[5]**

PROBLEM 2 [25]

Topics covered: conceptual design / ER modelling ; the transformation of an ER model into a relational model.

Question 1
Can a set of data requirements be *correctly* modelled by two or more different ER diagrams? Explain your answer. You may use a small example, if you think it will help your explanation. **[3]**

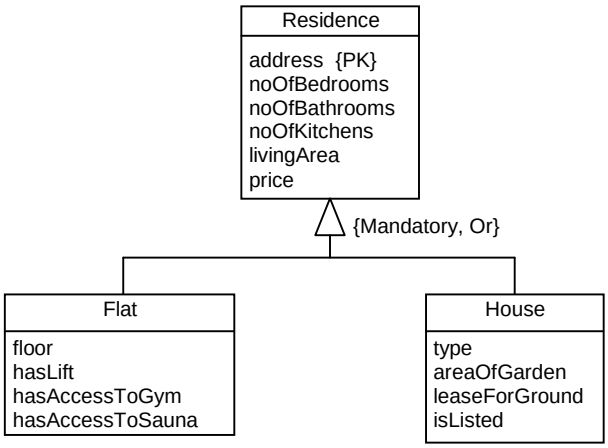
Question 2
Draw an ER diagram for the following description. Illustrate only the entity types (disregard the attributes), the relationships between them, and the multiplicity of each relationship. **[12]**

A company specialises on IT training. At the time being, the company has 20 instructors, provides 30 courses and can handle a maximum number of 600 trainees. However, these numbers may increase in the future. Each trainee registers for a minimum of 1 and a maximum of 3 courses. The number of trainees that can register for a course is not limited. Each course is assigned to a maximum number of 5 instructors. A course may be assigned to no instructors, if there are no trainees registered for it. An instructor may be assigned to a maximum of 10 courses. Each course is organised in 10 sessions. Each session is taught by one instructor, only. An instructor may be in charge of any number of sessions (obviously, an implicit constraint exists, namely that an instructor cannot be in charged of more than 100 sessions, but you may disregard this constraint).

Question 3
Draw an ER diagram for the following description. **[5]**

The students of a university register for different modules. One student may register for one or more modules (but not exceeding 24). One module, normally, has many students registered for it. If students fail a module they have to register again (they have to retake it). Therefore, the information relevant to registration is: date of registration and result.

Question 4
Consider the following ER diagram. Translate it into a relational model and specify the primary keys, foreign keys, and foreign key rules for each of the resulting relations. **[5]**



PROBLEM 3 [25]

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

Question 1
Explain how the process of normalisation can complement the process of ER modelling in database design. **[3]**

Question 2
Consider the following relation.

Student-Name	Username	Email	Course	Exam Date	Attempt	Result

(a) Give examples of three possible non-trivial functional dependencies (FDs) and concisely explain why did you consider them to be FDs. At least one FD should have a composite determinant. **[3]**

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

Question 3
Consider the following relation.

Patient	Disease	Doctor	Diagnosis	Treatment	Diet

and the following functional dependencies:

(Patient, Disease, Doctor) → Diagnosis
(Patient, Disease) → Treatment
Treatment → Diet

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). 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]**

Question 4
Consider the following relation R:

Patient	Disease	Doctor	Treatment

Consider the following functional dependencies for R:

(Patient, Disease) → Doctor
(Patient, Disease) → Treatment
Doctor → Disease

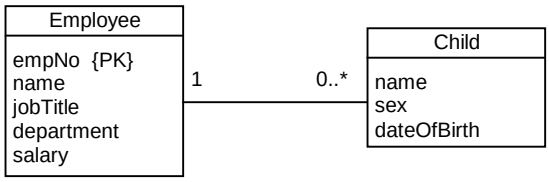
Assume they completely express all the functional dependencies existing in R. Discuss the way in which these functional dependencies can be expressed via normal forms (decomposition) in parallel with the issue of dependency preservation. **[6]**

PROBLEM 4 [25]

Topics covered: SQL (data definition, data manipulation, integrity constraints).

Question 1

Write the SQL statements that implement the database schema that corresponds to the following ER model. The entity "Child" is a *weak entity* which depends on "Employee". Your answer should include a statement of the relevant integrity constraints. The answer can be given purely in terms of two CREATE statements. **[6]**



Question 2

Express the following natural language queries in SQL (refer to the schema below):

(**a**) List the title, authors, and price for all the books published by Addison-Wesley in 2000, in alphabetical order with respect to titles. **[2]**

(**b**) List the titles of all the books that can be taken on loan for more than three days. **[2]**

(**c**) List how many non-returned books (as in physical copies) does the reader "Goldy Smith" have (a non-returned book has no value for 'dateIn') **[2]**

(**d**) List all the readers (as name and address) who have books overdue, together with the titles of these books – a book is considered overdue if it was not yet returned and it was on loan for more than the maximum number of days allowed ('maxDaysLoan'); (hint: assume that the difference between two values of type DATE corresponds to the data type associated with 'maxDaysLoan'; 'CURRENT_DATE' is an SQL unary operator which returns the current date). **[3]**

(**e**) List the names of all the readers who have non-returned books together with the total number of non-returned books, but only if this total exceeds their quota ('maxNoBooksForLoan'). **[4]**

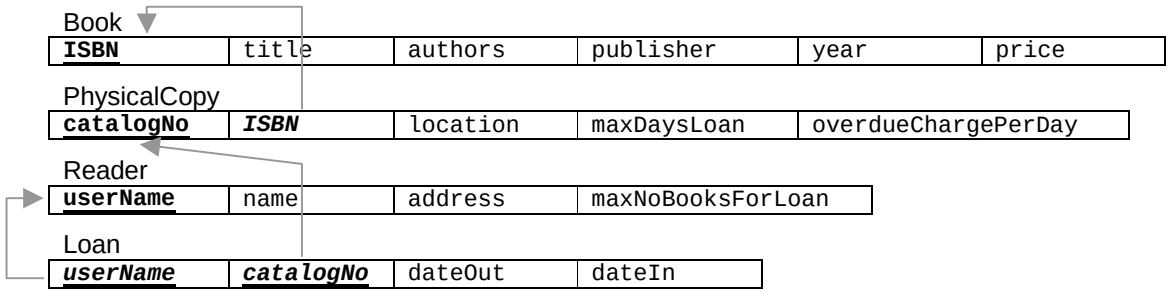
Question 3

Express the following integrity constraints in SQL (refer to the schema below):

(**a**) Books located in 'Reference' should not be allowed to be borrowed, i.e., the 'maxDaysLoan' for all their copies should be zero (note that this will not stop an actual loan to happen and even to be recorded in the database). **[3]**

(**b**) Books whose price exceeds £100 should not be allowed to be borrowed (the same observation as above applies here, too). **[3]**

Database schema for the above two questions:



PROBLEM 5 [25]

Topics covered: optimisation, transaction processing including recovery and concurrency.

Question 1

a) Explain, via a simple example, the idea of query optimisation. **[5]**

b) Enumerate some statistical information about a database that may be used by an optimiser. Where is such information stored? **[2]**

Question 2

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

b) Explain the ACID properties of transactions (one/two sentences per property). **[4]**

Question 3

There are five types of transactions that can be identified when a system failure arises. Describe each of them, stating, in each case, the corresponding recovery action that a DBMS must take (a diagram may help your explanation). **[5]**

Question 4

Consider the following transaction, called T, represented in Diagram 1 (t_i represent tuples). Explain the execution of T in time, in terms of locks, using the following primitives: **[5]**

request-for-lock(type, tuple); acquire-lock(type, tuple); wait; release-lock(type, tuple)

and the time scale represented in Diagram 2. Each horizontal line on the time scale could represent the execution of an operation, provided the requests for the corresponding locks is successful. The evolution of locks on these tuples from the point of view of another transaction, executed concurrently with T, is also described in Diagram 2.

```
BEGIN
   SELECT t1
   UPDATE t2
   UPDATE t3
   UPDATE t1
COMMIT
```

Diagram 1

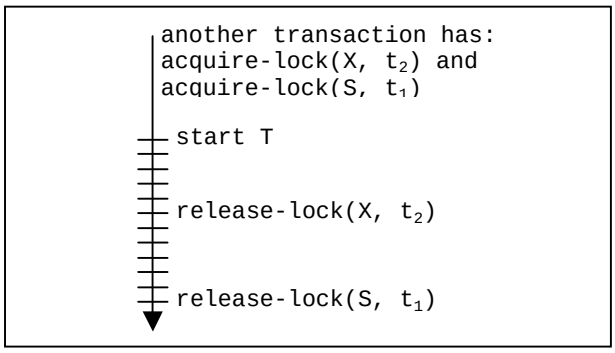


Diagram 2