The Open University

Mathematics and Computing: A Third Level Course
M358 Relational Databases

M358 SOLUTIONS TO SPECIMEN EXAMINATION PAPER

PART I

The following solutions for the Specimen Examination paper are given separately from the questions, but for the actual examination you must remember that your answers for Part I must be written in the spaces provided on the examination paper.

In some cases you should regard our answer as just one way of expressing an acceptable solution. Your answer, which may express the same meaning in a different way, would also be given full marks. Note that in Question 10(a) you are asked to give three answers; we have given five possibilities just so you can check your answer, but only three are required for the three marks. For some answers we have also added a remark in italics to explain the given answer, which would not be required in an examination (though note that some questions *do* require an explanation).

In your solutions to the examination questions, you should try to keep your answers as brief as possible, particularly for Part I. It is important to keep your answer focused on what is required by the question, and not add anything that is irrelevant or distracting (which may also be wrong!).

Question 1

(a) Information is any kind of knowledge that is exchangeable among people in some context. While data is a representation of information, it has no meaning without that context.

(b) (i) The department named Customer Support Services has a code of CSS11, a manager called John Sykes and employs 17 staff.

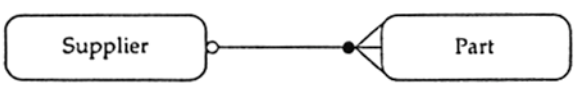
(ii) The value for Staff Employed is derived data because it is calculated from other data in the database.

Question 2

(a) 5 (*the number of lines*).

(b) The Supplies relationship is optional with respect to Supplier. An occurrence of Supplier may be related to many occurrences of Part via the Supplies relationship.

(c)



Question 3

(a) (*StudentId, CourseCode*)

(b) *StudentId* references *Student*
CourseCode references *Course*
TutorNo references *Staff*

(c) None

Question 4

(a) cardinality: 5, degree: 3.

(b) (i) 0 or 1
(ii) 60
(iii) 18 000

Question 5

(a) (i)

av_time
50

Give the average time committed to project p3 by employees assigned to it.

(ii)

project_no	*commited_time*
p1	20
p3	80

List the projects and time committed to those projects by the employee named 'Said'.

(b) Step 1 The FROM clause produces the Cartesian product of **assignment** and **employee**, giving an intermediate table.

Step 2 The resulting intermediate table is processed row by row, copying those rows that satisfy all of the conditions in the WHERE clause to a second intermediate table.

Step 3 The final table is formed from the second intermediate table by choosing only those columns named in the SELECT statement.

Question 6

(a) c1, c3, c4

(b) credit, c4

(c)

cx	*cy*	*cz*
75	c4	24
60	c3	null

Question 7

(a) `GRANT SELECT ON country TO PUBLIC`

(b) User **secretary** is given the UPDATE privilege for the **country** table, and the privilege to grant this to other users.

(c) The statement has no effect, as **secretary** does not have the privilege to grant INSERT access.

User **admin** will therefore only have SELECT privilege to the **country** table, as granted in part (a).

Question 8

An occurrence of A must be related to no more than one occurrence of B.
An occurrence of B must be related to exactly one occurrence of A.
An occurrence of B must be related to at least one occurrence of C.
An occurrence of C must be related to no more than one occurrence of B.
An occurrence of D must be related to exactly one occurrence of C.

Question 9

(a) $A,B \mapsto E$

(b) 2NF, because all the non-key attributes are fully-functionally dependent on the primary key, but there is a transitive dependency.

(c) *XBCNF (A, B, C, D)*
DBCNF (D, E)

Question 10

(a) Range of values
Precision
Operations on the values
Units of measurement
Scaling

(b) The CAST function enables any column (or generally an expression) to be transformed from one data type to another, so that the result of a query can be provided in whatever format is required by an application process.

PART II

Question 11

You were given the domains in the question and we reproduce them here to make it easier to follow the solution; you do not need to reproduce the domains in your answer. You are expected to give all the relations, including Member; in this case, nothing needs to be added to the Member relation given in the question, but in a similar question there might be a need to do so.

domains
MemberNumbers = integer
MemberNames = string
WalkDates = date
Routes = Integer
Descriptions = string
relation *Member*
MemberNo: MemberNumbers
Name: MemberNames
primary key *MemberNo*
relation *Walk*
WalkDate: WalkDates
Leader: MemberNumbers
Route: Routes
primary key *WalkDate*
foreign key *Leader* **references** *Member*
foreign key *Route* **references** *Route*
relation *Route*
RouteNo: routes
Description: Descriptions
primary key *RouteNo*
relation *Attendee*
MemberNo: MemberNumbers
WalkDate: WalkDates
primary key *(MemberNo, WalkDate)*
foreign key *MemberNo* **references** *Member*
foreign key *WalkDate* **references** *Walk*

(b) There are five mandatory participation conditions shown in the E–R model in Figure 3, represented as follows.

Attendee must participate in Is: foreign key MemberNo is already part of the primary key of Attendee *(so not null).*

Attendee must participate in AttendedBy: foreign key WalkDate is already part of the primary key of Attendee.

Walk must participate in Leads: requires addition to the foreign key Leader that it is not allowed null.

Walk must participate in UsedOn: requires addition to the foreign key RouteNo that it is not allowed null.

Walk must participate in AttendedBy: an explicit constraint is required as part of the Walk relation, as follows:

constraint
(project *Walk* **over** *WalkDate*)
difference
(project (*Attendee* **over** *WalkDate*)
is empty

The constraint is written over several lines just to make it easier for you to follow. You are not expected to explain how you arrive at your constraint but the following explanation may help you understand our solution.

> *The expression first finds a relation, that is, all the dates of walks – first line.*
> *It then subtracts (second line) all the dates where there was an attendee – third line.*
> *This relation must be empty – fourth line.*

(c)
constraint
(project *Walk* **over** *WalkDate, Leader*)
intersection
(project *Attendee* **over** *WalkDate, MemberNo*)
is empty

This constraint is expressed by finding the relation which gives any walk leader who is also an attendee on the same walk. This relation must be empty.

Question 12

(a) An SQL DECLARE section must be included to specify that *x*, *y* and *z* are host variables with appropriate (compatible) data types.

(b) For an input name of a city, give the name of the capital of the country to which the city belongs.

(c)
```
EXEC SQL
SELECT capital
INTO :z
FROM country
WHERE name =
    (SELECT country
    FROM city
    WHERE name = :y);
```

(d) The logical processing model for the nested query requires that the subquery is processed first; this involves using a copy of the **city** table, choosing the one row which has a value for **name** equal to the host variable *y* and then selecting the value for the column **country** as the result. The remainder of the nested query is processed by using a copy of the **country** table, choosing the one row which has a value for **name** equal to the result of the subquery and then selecting the value for the column **capital** as the result which is assigned to the host variable *z*.

This logical processing of the original two statements is essentially equivalent to the logical processing of the subquery. The result of the first query is then used in the logical processing of the second query, which is essentially equivalent to the logical processing of the main query using the subquery result.

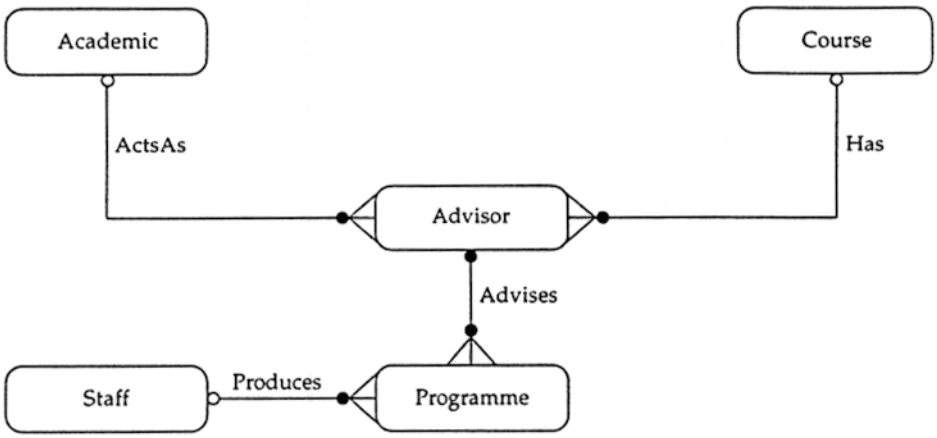
(e) If the values for the **name** column in the **city** table are not unique, there could be more than one row satisfying the condition in the first statement, but this form of embedded SQL statement can only return values for a single row. The embedded SQL statements in such a case may be described as follows:

Declare a cursor based on the same query specification as the first of the specified statements.

Input a value for *y*
Open the cursor
While there are rows to process
    Fetch the next row of the cursor table to give value for *x*
    Execute the second specified statement, to give a value for *z*
    Print *z*
End

Question 13

(a)



Entity types

Academic (Name, TelephoneNo)
Course (CourseCode, Title)
Advisor (Name, CourseCode, Contribution)
Programme (ProgrammeCode, Title, RunningTime)
Staff (StaffNo, Name, TelephoneNo)

Constraints

No additions necessary in this model

Assumptions

None

(b) (i) Entity subtypes are of no benefit because they should not be used in this situation; staff would have to be represented by an occurrence of both producer and director.

(ii) Entity subtypes would be beneficial because they avoid the need for an attribute, such as Position, to distinguish the positions and enable the constraint involving this attribute and the relationships with Programme to be represented directly rather than as a description.