BCS THE CHARTERED INSTITUTE FOR IT

BCS HIGHER EDUCATION QUALIFICATIONS
BCS Level 5 Diploma in IT

DATABASE SYSTEMS

Friday 1st April 2016 – Morning
Answer **any** FOUR questions out of SIX. All questions carry equal marks
Time: TWO hours

Answer any Section A questions you attempt in Answer Book A
Answer any Section B questions you attempt in Answer Book B

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

Calculators are **NOT** allowed in this examination.

Section A
Answer Section A questions in Answer Book A

A1

EXAMINER'S GENERAL COMMENTS

Almost half of candidates attempted this question. The average mark was disappointing and there was a wide distribution of marks.

It was surprising to see a relatively poor performance by many candidates as this question tested a well-covered area of the syllabus. - the principles of data definition, data input and integrity. The question required an application of knowledge to a practical situation. It was apparent that many candidates lacked experience in working with the SQL language.

Assume the following SQL script is to be executed :-

```
CREATE TABLE Students (StudentID CHAR(6)
                ,StudentFname VARCHAR(20) NOT NULL
                ,StudentLname VARCHAR(20) NOT NULL
                ,Date_of_Birth DATE
                ,CourseCode CHAR(6) ,
CONSTRAINT student_pk PRIMARY KEY (Studentid));

CREATE TABLE Courses(CourseCode CHAR(6) PRIMARY KEY
              ,CourseLevel INT
              ,Fee NUMBER(5,2));

INSERT INTO Students VALUES
('234349','Bill','Nomas', NULL,NULL);

INSERT INTO Students VALUES
('234350','Ramesh', 'Haslam', 22,'SET');
```

```
INSERT INTO Students VALUES
('234351','John','Norman', 24, 'GHR');

INSERT INTO Students VALUES
('234347','John','Sagatara', NULL, NULL);

INSERT INTO Students VALUES
('234350','Ramesh','Bartok', 22, NULL);

INSERT INTO Students VALUES
('234341','John','Norman', 24, NULL);

INSERT INTO Students VALUES
('234348', 'David', 'Bulmar', NULL, NULL);

INSERT INTO Students VALUES
('234345', NULL, 'Desai', NULL, NULL);

INSERT INTO Courses VALUES
('GHR',1,249 );
```

a) Classify TWO types of data integrity checks that have been specified to prevent invalid data being entered. Give examples of each of these types of data integrity checks.

(4 marks)

ANSWER POINTER

Entity Integrity ; CONSTRAINT student_pk PRIMARY KEY (Studentid)

Domain Integrity NULL Data Type range including characteristics of char and varchar

StudentID char(6) NOT NULL would restrict the input to 6 fixed chars and NULL values not allowed

EXAMINER COMMENTS (part a)
Most candidates could identify two different types of data integrity checks but this was meant to be in the context of the supplied script i.e. when data is input. It was a concern that a significant number of candidates reported Referential Integrity as a type of constraint that was present when it is clear the script does not implement this constraint!

b) What is the result of running the above script (assuming it is syntactically correct) in terms of table creation and insertion of data?

(2 marks)

ANSWER POINTER

Two tables created followed by successful INSERTS into table Students up to the INSERT

INSERT INTO Students VALUES

('234350','Ramesh', 'Bartok', 22);

PRIMARY KEY VIOLATION

There is also a NOT NULL violation

Leaving one row in the courses table and 6 rows in the students table

EXAMINER COMMENTS (part b)

The script is syntactically correct but generates an error when executed caused by a Primary Key (entity integrity) violation.

About half of candidates failed to see the Primary Key(PK) violation when the statement

```
INSERT INTO Students VALUES
('234350','Ramesh','Bartok', 22, NULL);
```

was executed. Otherwise answers either included a listing of the resultant tables without spotting the error or simply listing the number of rows (which was expected). Either of these answers was considered to be acceptable but failing to recognise the PK violation was penalised.

c) Change the script to enforce an additional constraint that would be applied to restrict the data of birth of all students to be later than 01-jan-1998.

(2 marks)

ANSWER POINTER

Check constraint added to a column

```
ALTER TABLE students ADD CONSTRAINT ck_birthdate CHECK(date_of_birth>'01-jan-1998');
```

EXAMINER COMMENTS

Most candidates identified that a check constraint was required but it was surprising how many candidates could neither name nor code it. Full marks were awarded for identifying that this is a type of constraint and that there was a reasonable pseudo-coding attempt.

d) Change the script to enforce an additional constraint that would ensure data integrity between data referenced in both the Students and Courses Tables. Assume that a student can only attend one course at a time and a course may have many students.

(4 marks)

ANSWER POINTER

Add to CREATE TABLE student statement

Referential Integrity : candidates should apply a REFERENCES clause to the STUDENT table creation statement eg Foreign key CourseCode REFERENCES Courses,CourseCode or show an alter table statement.

ALTER TABLE students ADD CONSTRAINT fk_courses FOREIGN KEY (coursecode) REFERENCES courses(coursecode);

EXAMINER COMMENTS
Most candidates spotted that a referential integrity constraint was required but it was surprising how many candidates had difficulty coding it. Full marks were awarded for identifying this as a type of constraint and that there was also a reasonable pseudo-coding attempt.

e) Why is it necessary to constrain updates performed on referenced data in tables such as Courses and Students? Explain the measures available in SQL to constrain these updates.

(5 marks)

ANSWER POINTER

The ON actions in the foreign key specifications refer to the referenced primary key. For the ON DELETE CASCADE example, when a primary key in the Course table is deleted (i.e. when a row in Course is deleted), then the delete action is "cascaded" to the foreign key, and all matching rows in Student table are also deleted.

EXAMINER COMMENTS
Many candidates had difficulty writing any answers to this section, probably due to lack of knowledge of the CASCADE DELETE clause. There were some candidates who were fully informed about this concept and gained full marks.

f) Describe with the aid of example SQL code how data may be inserted into a table by selecting and copying data from one or more existing tables, hence avoiding the use of many INSERT statements

(4 marks)

ANSWER POINTER

```
INSERT INTO Students
    (StudentID
    ,StudentFname
    ,StudentLname
)
SELECT Staff_id, fname, lname
FROM staff
WHERE status = 'Probation'
```

EXAMINER COMMENTS
Again a lack of SQL experience in manipulating/copying data was apparent as the same comments apply as those reported in the previous section

g) Apart from containing sequences of SQL code in a script, describe other ways that SQL code can be stored, contained, encapsulated and run as a sequence of executable statements.

(4 marks)

ANSWER POINTER

Stored procedures and triggers with explanations and examples are required such as

```
CREATE PROCEDURE inserts AS
(INSERT ….
COMMIT
INSERT …
```

EXAMINER COMMENTS
Again there was a noticeable lack of SQL experience of stored procedures/functions as a way of containing SQL code for repeatable running with/without parameters. Stored procedures make it easier for error conditions to be checked and recovery in case of failure. So the same comments apply as those reported in the previous sections. In this section however there were some answers that related more to embedding SQL code in scripting languages such as PHP/JSP. Technically this could be considered as another means to achieve what was asked; therefore marks were awarded if candidates answered in this way.

A2

EXAMINER COMMENTS OVERALL

Just over half of the candidates attempted this question. The average mark was reasonable, and there was a wide distribution of marks with almost half of the attempts achieving a pass mark.

Observe the following scenario

*A travel company provides a selection of **Hotels** that prospective customers can reserve prior to booking a room**.** A customer can select from a range of **Accommodation Types** that each hotel offers to suit their requirements. Details of the accommodation type include the catering facilities either Self Catering (SC); Half Board (HB); Full Board (FB). The bed type either Twin bed (T); Double bed (D); Suite (S). The price of the hotel is determined by the hotel and the type of accommodation offered.*

Assume that

- *Each hotel is identified by a hotel code.*
- *The accommodation type is identified by a unique accommodation type code.*
- *Accommodation is only available during the month of June in 2016.*

Fig A2 below is a representative sample of data about hotels and the accommodation types offered by each hotel.

Fig A2 Hotel Accommodation

RESORT	HOTEL CODE	HOTEL NAME	CATERING	BED TYPE	ROOM PRICE	ACC_TYPE _CODE	MONTH
Benidorm	FLB	Flamingo	SC	T	159	12	June
Palma Nova	JDM	Jardin Del Sol	SC	T	195	12	June
Benidorm	AHB	Al Hambra	FB	D	199	15	June
Santa Ponsa	HAZ	Hawaii	HB	T	308	16	June
Playa Blanca	SPZ	Sun Park	FB	S	310	18	June
Benidorm	AHB	Al Hambra	HB	S	199	17	June
Palma Nova	JDM	Jardin del Sol	FB	S	199	18	June
Palma Nova	JDM	Jardin del Sol	FB	D	169	15	June
Benidorm	SPB	Sun Park	FB	S	159	18	June

a) Derive an Entity Relationship data model for the above scenario according to the following requirements:-

(You may use any stated standard ER modelling notation and you must state any assumptions necessary but do not contradict the scenario)

Entity Types that you model are listed in bold font

(2 marks)

Show Relationships and participation constraints

(5 marks)

Resolve Many to Many relationships

(4 marks)

Allocate attribute types to Entity Types using the column headers from Table A2. Underline attributes that are Entity Identifiers.

(4 marks)

ANSWER POINTER

ER model is supplied in Fig A2S with new Entity Types presented to resolve the Many to many relationship between *Hotel* and *Accommodation Type* renamed as Entity type Hotel Accommodation.

EXAMINER COMMENTS
Candidates are advised to always state the notation used in their ERD and make sure it is used accurately and consistently otherwise marks can be lost. The conversion of the many-to-many relationship to two one to many relationships is fairly standard and candidates are advised to study this carefully in the supplied solution.

b) Derive a set of Tables/Relations from your ER model containing the sample data above. Underline the Primary Keys in each Table.

(5 marks)

ANSWER POINTER

Hotel Table

HOTEL CODE	**HOTEL**	**RESORT**
FLB	Flamingo	Benidorm
AHB	Bali Hai	Benidorm
HAZ	Hawaii	Santa Ponsa
SPZ	Sun Park	Playa Blanca
AHB	Al Hambra	Benidorm
JDM	Jardin del Sol	Palma Nova
SPB	Sun Park	Playa Blanca

Hotel Accommodation (new Linking Table)

HOTEL CODE	ACCOM_TYPE_ID	PRICE	MONTH
FLB	12	159	June
JDM	12	195	June
AHB	15	199	June
HAZ	16	308	June
SPZ	18	310	June
AHB	17	199	June
JDM	18	199	June
JDM	15	169	June
SPB	18	159	June

AccommodationType Table

ACCOM_TYPE_ID	CATERING	BEDTYPE
12	SC	T
13	SC	D
14	SC	S
15	FB	D
16	HB	T
17	HB	S
18	FB	S

EXAMINER COMMENTS

The mapping to tables from an ERD was quite straightforward but this depended on section a) and how the relationship table (Hotel Accommodation) allocated attribute types to column names (note the allocation of price and month). Again some candidates failed to show data in the tables and lost marks as a result.

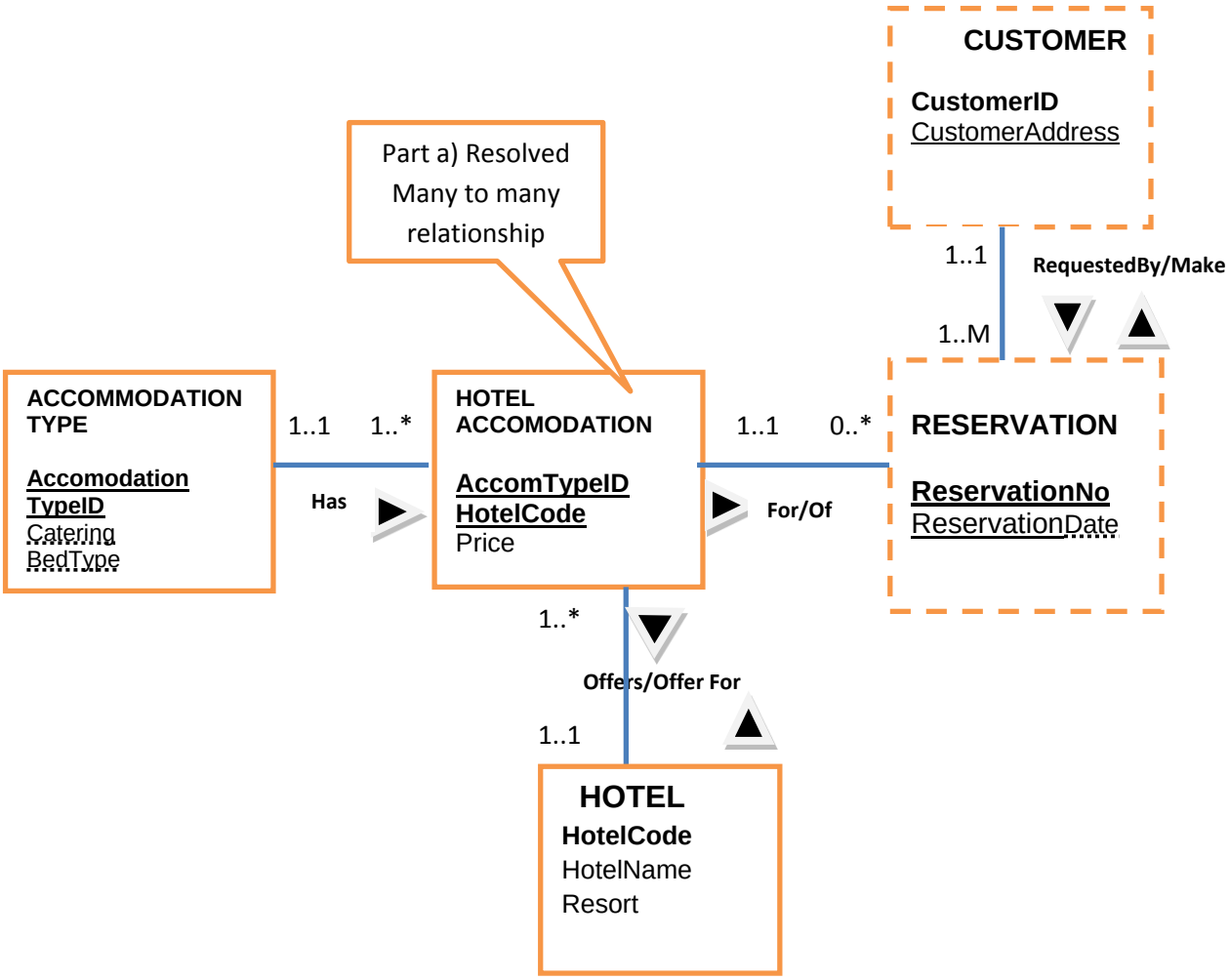
c) Extend the ER model you produced above in part a) by adding **TWO** further Entity Types; **Customer** and **Reservation**; to allow a Customer to reserve accommodation at a hotel possibly on different dates.

(5 marks)

ANSWER POINTER

ER model is supplied in Fig A2S with Entity Types Customer and Booking added in dotted lines.

Figure A2s Data model solution to question 2 parts a) and c) (dotted boxes)



EXAMINER COMMENTS

It should be noted that the link to Reservation is made with the Hotel Accommodation entity type rather than to the Hotel entity type as some candidates thought. This is because Hotel Accommodation is a room type in a particular hotel which a customer could reserve. However some candidates translated this further into a allocated room booking; again this was unnecessary as the question did not require this extension to the discourse. Overall other assumptions were not considered unless there were clearly stated assumptions that did not contradict the discourse.

A3

EXAMINER OVERALL COMMENTS

Almost all candidates attempted this question covering a well-covered and familiar topic of normalisation. There was a wide distribution of marks with around two thirds of candidates achieving a pass mark.

(a) A library uses the following table to store details of students, the books they have borrowed and when they borrowed them. The Primary Key is (StudentID, BookID).

Borrowing

StudentID	**StudentName**	**BookID**	**BookTitle**	**Date**
S1	Smith	B1	Python	12-Apr-2016
S1	Smith	B2	Databases	17-Jan-2016
S2	Ford	B1	Python	25-Feb-2016

(i) Which Normal Form does the above table violate and why?

(3 marks)

ANSWER POINTER

The table violates 2nd Normal Form (1 mark)

- because there are two partial dependencies:
  StudentID → StudentName and BookID → BookTitle (2 marks)

(ii) Give an example of an insert anomaly and an example of a delete anomaly that may occur if the table is left un-normalised.

(4 marks)

ANSWER POINTER

Insert anomaly: a new book cannot be added without having a student borrower associated with it (2 marks). Alternatively: when inserting a new record, the title of an existing book is mistyped creating multiple copies for the same book.

(iii) Normalise the table to achieve 3rd normal form.

(5 marks)

ANSWER POINTER

Student(StudentID, StudentName) (1 mark)
Book(BookID, BookTitle) (1 mark)
Borrowing(StudentID*, BookID*, Date) (3 marks)

EXAMINER COMMENTS
Most candidates performed well on this section. However candidates are advised to show working out in particular to show the steps involved in producing a set of normalised tables In (part a) iii), marks were still awarded if an intermediate step was correct but the result was incorrect.

(b) An important concept in the theory of relational databases is that of a *functional dependency*.

(i) Explain what is meant by a functional dependency and give an example.

(2 marks)

ANSWER POINTER

An attribute B is functionally dependent on an attribute A if each value of A is associated with a value of B (1 mark). For example, StudentID → StudentName (1 mark).

(ii) Identify two functional dependencies in the following table (A, B and C are the names of the columns):

A	***B***	***C***
a1	b1	c1
a1	b1	c3
a1	b2	c1

(4 marks)

ANSWER POINTER

B → A (2 marks) and C → A (2 marks)

EXAMINER COMMENTS
It was surprising to see a disproportionate number of incorrect answers to part b) ii).

(c) A football club uses the table below to record details of players and the positions in which they can play. Each player can play in up to a maximum of three positions:

playerID	**playerName**	**positions**
P1	Lionel Messi	Forward, Centre Midfield
P2	Cristiano Ronaldo	Forward, Left Midfield, Right Midfield
P3	Philippa Lahm	Right Back, Defensive Midfield

(i) Explain why this table is not in “First Normal Form” (1NF).

(1 mark)

ANSWER POINTER
The attribute “positions” is multi-valued on the PK, hence the table is not in 1NF. (1 mark)

(ii) Show how this table can be transformed into 1NF tables. Give **two** possible solutions.

(6 marks)

ANSWER POINTER
Solution 1:
players (playerID, playerName, position1, position2, position3) (2 marks)

Solution2:
players (playerID, playerName) (2 marks)
positions (playerID*, position) (2 marks)
Alternatively, can use a third table (positionID, positionName)

Also Player_position_id,player_id,name,position

EXAMINER COMMENTS
Overall this question was quite well answered with most candidates specifying solution2. Many candidates could produce only a slight variation of solution2 as an alternative solution and not what was expected.

Section B
Answer Section B questions in Answer Book B

B4

(a) Relational Algebra (RA) consists of two sub-categories of operation:

- Those based on set theory and essentially borrowed from mathematics
- Those invented specifically for the manipulation of relations (not present in set theory)

For EACH sub-category, name, describe, draw and illustrate (using your own examples) THREE separate RA operations, making a total of SIX operations. Good diagrams will gain extra credit.

Each operation is worth up to three marks.

(18 Marks)

ANSWER POINTER

This will be marked at 3 marks per operation *but with extra marks for good examples and clear diagrams*. Good answers should cover: CARTESIAN PRODUCT (TIMES), UNION and UNION ALL, INTERSECTION AND MINUS (DIFFERENCE) operators for the first category, with SELECTION, PROJECTION, JOIN, and DIVIDE belonging to the latter category. For each operator, either a Venn diagram or other conceptual illustration and suitable RA example is expected.

(b) Those data sets operated upon by operations drawn from set theory have to abide by a particular criterion that data sets being manipulated by the other operations do not. What is this criterion and what restrictions does it place upon the relevant data sets?

(7 Marks)

ANSWER POINTER

A clear definition of *union compatibility* - stating that in order to apply the set operators all component sets must have the same degree (number of columns) and that comparable columns in each set must be of the same data type (compare like with like). It should also be made clear that UNION (an operator to be applied) is not the same as UNION COMPATIBILITY (a condition to be satisfied).

EXAMINER COMMENTS

Generally, very well answered – especially section (a). This is well-trodden material and it should come as no surprise that the vast bulk of candidates have a firm grip on these fundamental relational concepts. Most supplied a clear explanation, Venn diagram and example relations. In section (b), most had the correct idea but a small minority continue to confuse union compatibility with the union (and other) set operators. Overall, a strong question for the bulk of students who attempted it.

B5

(a) The database is only one component in a larger IT infrastructure. Using your own simple examples and any diagrams you feel suitable, explain how
   1) The 2-Tier and 3-Tier architectures work
   2) The interfaces differ across those architecture
   3) The split between the traditional data and logic tiers works within these environments.

(15 Marks)

ANSWER POINTER

To be marked holistically but a good response will address the following: how 2-Tier are client-server and 3-Tier are web-based architectures. How in a 2-Tier model, the client handles essentially the presentation work while the back-end database looks after the logic and data management aspects. In a 3-Tier approach, the browser tends to focus on presentation, the database the data management and the intervening middleware/web server the business logic. The better students will then include issues such as (non-technical) end-user interfaces like web-based (three-tier) interfaces, non-web (two-tier) GUIs such as forms & reports through to those interfaces designed for technical users like developers and DBAs such as forms/report generators and other software development environments – both graphical and command-line (for example Oracle's APEX and SQL*Plus respectively). It also includes third party interfaces for developers such as TOAD and specialized applications like Oracle's Enterprise Manager for DBAs. It would also be nice to see comments regarding interactive use versus scripted or programmatic usage.

b) Explain what the term *data validation* means. Using your own examples, describe the various data validation techniques that may be embedded into a forms-based interface to a database – for example, ensuring that the correct type and range of data values are entered.

(10 Marks)

ANSWER POINTER

The definition should be along the lines of 'ensuring that only clean, correct, accurate, well-formatted data is accepted into the database'. It should be about data integrity, not confused with database security (although validating a user ID and password is fine here). Typical examples will include format masks (for example, dates of birth or telephone numbers have valid structure), range checks (for example, financial fields are > 0.00), membership validity (for example, a supplier name is a valid entry in the database), cross-field consistency checks (so if the user enters 'Mr' in one field, the Sex field must be 'M' etc.), presence checks (to ensure no missed fields) etc. Clear examples needed for best marks.

EXAMINER COMMENTS

Answers to section (a) were quite variable in terms of quality - ranging from extensible and detailed responses, supported by quality architecture diagrams to superficial and vague answers lacking any diagrams. By contrast, section (b) was generally very well answered by most candidates – with most grasping all the key issues outlined in the marking scheme.

B6

(a) Explain what is meant by a transaction and why it is an important unit of operation in a DBMS?

(2 marks)

ANSWER POINTER
Transaction: An action, or series of actions, carried out by a single user or application program, which reads or updates the contents of the database. A logical unit of work that transforms the database from one consistent state to another. (1 mark)

Its importance stems from the fact that it is the unit of concurrency and recovery control in a database system. (1 mark)

EXAMINER COMMENTS

Answered surprisingly poorly by many candidates with a range of vague and rambling responses. Some did get the core ideas of a *logical unit of work*, *abiding by the ACID principle*, that transfers a database from *one consistent state to another* – but most did not. Some talked about financial transactions.

(b) Suppose we have a table called students that, initially, has 120 records. **How many rows** will be in the table after executing the following commands? **Justify** your answer. Hint: the answer is one of the following: 0, 1, 122 or 123.

```
INSERT INTO students (stud_id) VALUES (120);
SAVEPOINT stud120;

INSERT INTO students (stud_id) VALUES (121);
SAVEPOINT stud121;

INSERT INTO students (stud_id) VALUES (122);
SAVEPOINT stud122;

TRUNCATE TABLE students;

INSERT INTO students (stud_id) VALUES (123);

ROLLBACK;
```

(4 marks)

ANSWER POINTER
There will be 0 records in the students table. The TRUNCATE statement is DDL and performs an implicit commit. After the TRUNCATE statement, there are 0 rows in the table. The one row that was inserted was removed when the ROLLBACK statement was executed.

EXAMINER COMMENTS

Again, generally poorly answered with only a minority of candidates grasping the role played by the TRUNCATE statement – although more identified the impact of the final ROLLBACK. Most students did not identify 0 (zero) rows as the correct answer.

(c) Describe, with an example, one type of problem that can occur in a multi-user environment when concurrent access to the database is allowed. **(6 marks)**

ANSWER POINTER
2 marks for description + 4 marks for example of one of the following: Lost update problem, the uncommitted dependency problem, and the inconsistent analysis problem.

EXAMINER COMMENTS

Most candidates clearly discussed the concept of concurrency and potential contention between parallel transactions but only around half then went on to describe specific situations – like the lost update problem in detail.

(d) Backups of the database should be taken in order to protect data. Describe **five** measures that can be taken in order to ensure the security and effectiveness of database backups.

(5 marks)

ANSWER POINTER
Examples of measures (1 mark each)

- Encrypt data in the backup.
- Take multiple copies.
- Create copies on different media (e.g., disks, tapes)
- Store in a site different from the database site.
- Test and validate the backups (trial recoveries).

EXAMINER COMMENTS
A much better answered question with most candidates making a series of valid points.

(e) Describe four possible benefits of "**Views**" in databases.

(8 marks)

ANSWER POINTER
(2 marks each)

- Reduce complexity: for example tables could be joined and presented as part of one view
- Provide a level of security: sensitive data could be removed from a view, and users given access to the view instead of the base table
- Provide a mechanism to customize the appearance of the database: for example columns could be renamed in order to present a customized view to each end user
- Present a consistent, unchanging picture of the structure of the database, even if the underlying database is changed

EXAMINER COMMENTS
Again, mostly well answered.