OCR
RECOGNISING ACHIEVEMENT

ADVANCED GCE   **2509**

COMPUTING

Systems Software Mechanisms, Machine Architecture,
Database Theory and Programming Paradigms

WEDNESDAY 23 JANUARY 2008   Morning

                                  Time: 1 hour 30 minutes

Candidates answer on the question paper.
Additional materials: No additional materials are required

Candidate
Forename

Candidate
Surname

Centre
Number

Candidate
Number

INSTRUCTIONS TO CANDIDATES

* Write your name in capital letters, your Centre Number and Candidate Number in the boxes above.
* Use blue or black ink. Pencil may be used for graphs and diagrams only.
* Read each question carefully and make sure that you know what you have to do before starting your answer.
* Answer **all** the questions.
* Do **not** write in the bar codes.
* Do **not** write outside the box bordering each page.
* Write your answer to each question in the space provided.
* Additional answer space is available on the lined pages at the back of this booklet. Answers on these pages **must** be clearly numbered.

INFORMATION FOR CANDIDATES

* The number of marks for each question is given in brackets [ ] at the end of each question or part question.
* The total number of marks for this paper is **90** of which 4 marks are allocated to the assessment of the quality of written communication.
* No marks will be awarded for using brand names of software packages or hardware.

FOR EXAMINER'S USE	
1	
2	
3	
4	
5	
6	
7	
8	
QWC	
TOTAL	

This document consists of **12** printed pages, **3** lined pages and **1** blank page.

     [Turn over

Answer **all** questions.

1   Program P and Program Q are both ready for the processor. Program P has a higher priority than Program Q. While processing these programs, interrupts will occur.

   (i)   Explain the term priority.

   ....

   ....

   ....

   ....[2]

   (ii)   Explain the term interrupt.

   ....

   ....

   ....

   ....[2]

   (iii)   Describe how the processor will handle these two programs.

   ....

   ....

   ....

   ....

   ....

   ....

   ....

   ....[4]

2 **(a)** Explain the purpose of a compiler.

....

....

....

....[2]

(b) Describe what happens during syntax analysis.

....

....

....

....

....

....

....

....[4]

(c) Instead of a compiler, an interpreter may be used during program development.

Explain how errors are handled by using an interpreter.

....

....

....

....

....

....

....

....[4]

[Turn over

3 **(a)** Describe Von Neumann architecture.

....

....

....

....

....

....[3]

(b) **(i)** State **three** phases of the cycle performed by the processor when running a program.

1 ....

2 ....

3 ....[3]

(ii) Explain the term parallel processing.

....

....

....

....[2]

(iii) Explain **one** advantage and **one** disadvantage of parallel processing.

Advantage ....

....

....

....

Disadvantage ....

....

....

....[4]

4 (a) The following data are to be stored, in the order given, into a binary sort tree.

dog   rabbit   cat   fish   turtle   mouse

Draw a diagram to show the tree obtained.

[3]

(b) Show the stages of an insertion sort to put the numbers in ascending numerical order. The routine has been started for you.

Instruction   Numbers

	5	6	2	9	3	17	1
No change:	5	6	2	9	3	17	1
Insert 2:	2	5	6	9	3	17	1

[4]

[Turn over

5 When developing a program in a high level language, stepwise refinement may be used.

(a) Explain the term stepwise refinement.

....

....

....

....[2]

(b) Explain why procedures may be used in developing a program.

....

....

....

....

....

....[3]

(c) Explain how a stack is used when a procedure is called by a program.

....

....

....

....

....

....

....

....[4]

7

BLANK PAGE

PLEASE DO NOT WRITE ON THIS PAGE

PLEASE TURN OVER FOR THE NEXT QUESTION

6 **(a)** The following statements may apply to a data description language (DDL), a data manipulation language (DML) or both.

Tick **one** box in each row to show whether the statement applies to DDL, DML or both.

Statement	DDL only (✓)	DML only (✓)	Both (✓)
It is used with databases.			
It is a high level language.			
It is used to create new tables.			
It can query data.			
It is used to define primary keys.			
It can update existing data.			

[6]

(b) Explain **two** advantages of using a relational database compared with flat files.

Advantage 1 ....

....

....

....

Advantage 2 ....

....

....

....[4]

(c) A relational database is used to store data in a health centre. Part of the entity-relationship (E-R) diagram is shown.

| Patient |━━━◁| Appointment |▷━━━| Doctor |
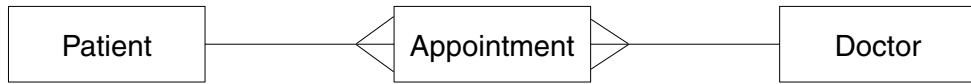
(i) State the type of relationship between Patient and Appointment in the diagram.

....[1]

(ii) Describe the use of primary and foreign keys in the Patient and Appointment tables.

....

....

....

....

....

....[3]

(iii) State **one** reason why the E-R diagram was **not** given as

| Patient |▷━━━◁| Doctor |

....
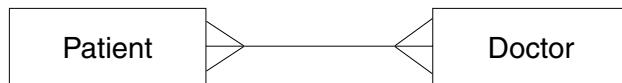
....[1]

(iv) State the meaning of the term secondary key and explain **one** example of the use of a secondary key in the Patient table.

....

....

....

....

....

....[3]

　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　　[Turn over

7 (a) The diagram shows a queue containing three data items.

Mary	Tom	Anu	

↑ front   ↑ rear
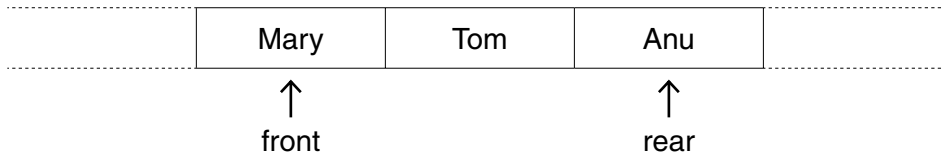
(i) Draw a diagram to show the queue when the data item Jay is added.

[2]

(ii) Draw a diagram to show the queue when **two** data items are removed from the **original** queue.

[1]

(b) (i) State the main feature of a stack data structure.

....

....[1]

(ii) An algorithm for adding one new data item to a stack could be

```
begin
        if stack_is_full
                then output "Stack full"
        else
        begin
                stack_pointer := stack_pointer + 1
                data (stack_pointer) := new_data
        end
end
```

Using the same style, write an algorithm for removing one data item from a stack and displaying a message of the form "Item removed was …".

....

....

....

....

....

....

....

....

....

....[5]

(c) (i) State the difference between static and dynamic data structures.

....

....[1]

(ii) State **two** advantages of using a static data structure.

Advantage 1 ....

....

Advantage 2 ....

....[2]

[Turn over

8 Some relationships are defined using a declarative language.

female (jackie)   {Jackie is female}
female (emma)
male (omar)   {Omar is male}
male (harry)
likes (tom, omar)   {Tom likes Omar}
likes (tom, jackie)
likes (jackie, emma)
likes (omar, tom)
likes (emma, harry)

Rules for combining these are

are_friends (X,Y) if
likes (X,Y) and
likes (Y,X)

is_girlfriend (X,Y) if
female (X) and
likes (Y,X)

(a) State why

   (i) ? are_friends (tom, omar) = true

   ....

   ....

   ....[1]

   (ii) ? is_girlfriend (emma, harry) = false

   ....

   ....

   ....[1]

(b) The user wants to find all values where is_girlfriend (X, tom)

    (i) State **one** goal.

        ....

        ....[1]

    (ii) State **one** example of instantiation.

        ....

        ....[1]

    (iii) Showing how you obtain your answer, list the value(s), if any, that the user should find where

        is_girlfriend (X, tom)

        ....

        ....

        ....

        ....

        ....

        ....

        ....

        ....

        ....

        ....

        ....[6]

If you use these lined pages to complete the answer to any question, the question number **must** be clearly shown.

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

[Turn over

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....