OXFORD CAMBRIDGE AND RSA EXAMINATIONS

Advanced GCE

COMPUTING   **2509**

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

Tuesday   **20 JUNE 2006**   Morning   1 hour 30 minutes

	Candidate Name		Centre Number	Candidate Number

TIME   1 hour 30 minutes

INSTRUCTIONS TO CANDIDATES

• Write your name in the space above.

• Write your Centre Number and Candidate Number in the boxes above.

• Answer **all** the questions.

• Read each question carefully and make sure you know what you have to do before starting your answer.

• If you run out of space for an answer, continue on the spare pages at the back of the booklet.

• If you use these spare pages, you must write the question number next to your answer. You can also use these spare pages for rough work.

INFORMATION FOR CANDIDATES

• The number of marks is given in brackets [ ] at the end of each question or part question.

• The total number of marks for this paper is 90 (86 + 4 written communication).

• You will be awarded marks for the quality of written communication where an answer requires a piece of extended writing.

• No marks will be awarded for using brand names of software packages or hardware.

FOR EXAMINER'S USE		
Qu.	Max.	Mark
1	14	
2	13	
3	13	
4	9	
5	13	
6	10	
7	14	
WC	4	
TOTAL	90	

This question paper consists of 11 printed pages, 2 lined pages and 3 blank pages.

1 **(a)** When using virtual memory, disk threshing may occur.

    (i) Describe virtual memory and explain why it may be needed.

....

....

....

....

....

....

....

....[4]

    (ii) Explain the term *disk threshing*.

....

....

....

....

....

....[3]

(b) Explain why the contents of memory need to be managed.

....

....

....

....

....

....[3]

(c) For a network operating system, explain the term *transparency*.

....

....

....

....[2]

(d) Explain why a job with low priority may have its priority changed by the operating system.

....

....

....

....[2]

2 Programs written in high-level languages must go through a variety of translation processes before they can be executed.

(a) Lexical analysis occurs during compilation.

State **two** other stages of compilation.

1. ....

....

2. ....

....[2]

(b) Describe the process of lexical analysis.

....

....

....

....

....

....[3]

[Turn over

(c) Explain the term

    (i) executable code.

       ....

       ....

       ....

       ....[2]

    (ii) interpreter.

       ....

       ....

       ....

       ....[2]

(d) Some compilers change the source code to intermediate code instead of executable code. The intermediate code is then run using an interpreter.

    (i) When this method is used, state at what stage all the syntax errors are found and corrected, giving a reason for your answer.

       ....

       ....

       ....

       ....[2]

    (ii) Explain **one** disadvantage of this method compared to using a compiler to produce executable code.

       Disadvantage ....

       ....

       ....

       ....[2]

3 **(a)** Describe the purpose of the memory data register.

....

....

....

....

....

....[3]

(b) Interrupts are given priorities.

(i) Explain the term *interrupt*.

....

....

....

....[2]

(ii) Explain why priorities are needed for interrupts.

....

....

....

....

....

....[3]

(c) At a particular stage in a program, the program counter holds the value 123. Its next value is 38. No interrupt has occurred. (Numbers are given in denary to help you.)

Explain what is happening in the program.

....

....

....

....[2]

[Turn over

(d) State the main features of Von Neumann architecture.

....[3]

....

....

....

....

....[3]

4 **(a)** **(i)** Explain the differences between a queue and a stack in a computer system.

....

....

....

....

....

....

....

....

....[4]

(ii) Give **one** example of the use of each of these structures.

Queue ....

....

Stack ....

....[2]

(b) Using a diagram, describe how a stack may be used to reverse the order of items held in a queue.

....

....[3]

5 **(a)** State **three** characteristics of a low-level language.

1. ....

....

2. ....

....

3. ....

....[3]

(b) **(i)** A particular low-level language uses the instruction ADN 23 to mean "add the number 23 to the number in the accumulator".

Name this type of addressing.

....[1]

(ii) Another instruction ADD 23 means "add the number in address 23 to the number in the accumulator".

Name this type of addressing.

....[1]

(iii) State the purpose of the accumulator.

....

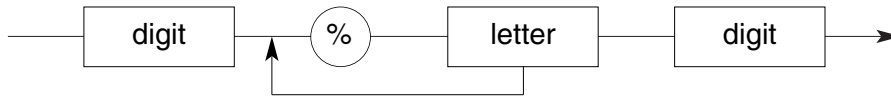...[1]

(c) Describe the differences between procedural and declarative high-level languages.

....

....

....

....

....

....

....

....[4]

[Turn over

(d) In a particular language, "value" is defined by the following syntax diagram:

value:

State whether each of the following is a value or not, giving a reason in each case.

(i) 2 % A 3

....

....[1]

(ii) 5 % B % % C % D 6

....

....[1]

(iii) 7 % E % 8

....

....[1]

6 **(a)** **(i)** Write a short algorithm for the inorder traversal of a binary tree.

....

....

....[3]

(ii) Explain the term *recursion*.

....

....

....

....

....

....[2]

(b) The binary tree shows some children's names sorted into alphabetical order.

```
                        Peter
            Jai                        Sarah
                Jessica        Rosita        Tom
```
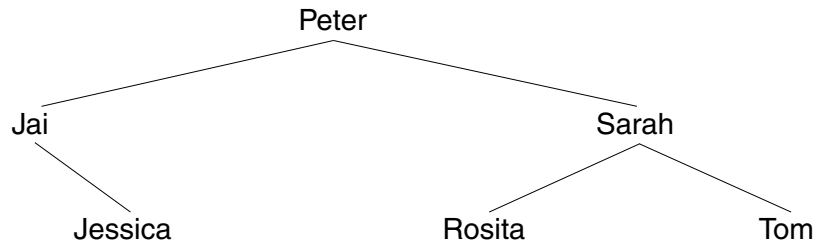
(i) Insert the names Maggie and Sumeer into the sorted binary tree below.

```
                        Peter
            Jai                        Sarah
                Jessica        Rosita        Tom
```
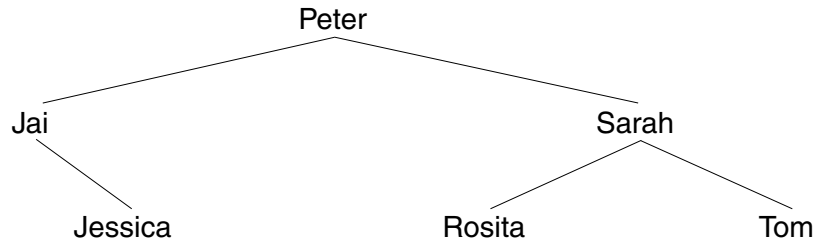
[2]

(ii) Starting from the original binary tree given, draw a new version of the diagram after the name Sarah is removed.

[1]

(c) Describe how an item is removed from a binary tree.

....

....

....

....[2]

[Turn over

7 Data for a large hotel is stored in a relational database.

(a) Part of the data model is shown:

```
┌─────────┐        ┌─────────────┐        ┌─────────┐
│  Guest  │────◁───│ Reservation │───▷────│  Room   │
└─────────┘        └─────────────┘        └─────────┘
```

Guest (<u>GuestCode</u>, Surname, Forename)
Room (<u>RoomNo</u>, Cost)
Reservation (<u>GuestCode, RoomNo, StartDate</u>, EndDate, NoOfPeople)

For example, a Reservation could show that a guest with code G1004 booked room number 123 from 25 June 2006 to 27 June 2006 for 2 people.

(i) State the degree of relationship between Guest and Reservation.

....

....[1]

(ii) Identify **one** foreign key from this data model.

....

....[1]

(iii) Explain the use of this foreign key.

....

....

....

....[2]

(b) (i) Explain the relationship shown in the E-R diagram below.

```
┌─────────┐        ┌─────────┐
│ Cleaner │───▷──◁───│  Room   │
└─────────┘        └─────────┘
```

....

....

....

....
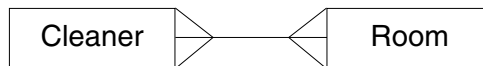
....[2]

(ii) Draw a new version of this E-R diagram in third normal form.

[3]

(c) (i) State why the cleaning supervisor should be given access to the Reservation data.

....

....[1]

(ii) State why this access should be read-only.

....

....[1]

(d) The reservation staff have read/write access to the data.

Explain how the system will handle the situation when two members of staff attempt to access the same data.

....

....

....

....

....

....[3]

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

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

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

14

BLANK PAGE

15

BLANK PAGE

BLANK PAGE