OCR
RECOGNISING ACHIEVEMENT

GCE

# Computing

Advanced GCE **A2 7820**

Advanced Subsidiary GCE **AS 3820**

# Combined Mark Schemes And Report on the Units

## June 2005

**3820/7820/MS/R/05**

Oxford Cambridge and RSA Examinations

OCR (Oxford, Cambridge and RSA Examinations) is a unitary awarding body, established by the University of Cambridge Local Examinations Syndicate and the RSA Examinations Board in January 1998. OCR provides a full range of GCSE, A level, GNVQ, Key Skills and other qualifications for schools and colleges in the United Kingdom, including those previously provided by MEG and OCEAC. It is also responsible for developing new syllabuses to meet national requirements and the needs of students and teachers.

This mark scheme is published as an aid to teachers and students, to indicate the requirements of the examination. It shows the basis on which marks were awarded by Examiners. It does not indicate the details of the discussions which took place at an Examiners' meeting before marking commenced.

All Examiners are instructed that alternative correct answers and unexpected approaches in candidates' scripts must be given marks that fairly reflect the relevant knowledge and skills demonstrated.

Mark schemes should be read in conjunction with the published question papers and the Report on the Examination.

OCR will not enter into any discussion or correspondence in connection with this mark scheme.



Any enquiries about publications should be addressed to:

OCR Publications
PO Box 5050
Annersley
NOTTINGHAM
NG15 0DL

Telephone: 0870 870 6622
Facsimile: 0870 870 6621
E-mail: publications@ocr.org.uk

## CONTENTS

### Advanced GCE Computing (7820)

### Advanced Subsidiary GCE Computing (3820)

### MARK SCHEMES FOR THE UNITS



### REPORT ON THE UNITS

# Mark Scheme 2506
# June 2005

**1** **(a)** **(i)** (The set of programs needed to) control the hardware of a computer/software **[1]**

**(ii)** A program that allows the user to produce something worthwhile/to carry out a task/suitable example **[1]**

**(iii)** System programs (designed to carry out a common task)/suitable example **[1]**

**(iv)** Provides a means of communication between the user and the computer **[1]**

**(b)** -Computer cannot execute HLL/translator changes code in HLL/into a code that the computer can execute **[1]**

**2** **(a)** -LAN over short distance/WAN over long distance
-LAN hardwired (or equivalent)/WAN needs other form of communication
-WAN requires modem (or other)/LAN does not
(1 per -, max 2) **[2]**

**(b)** Any situation where it would be reasonable to hard wire/where computers are in one site
e.g. school computer room **[1]**
Any situation where hard wiring unlikely because of distance
e.g. Branches of a store across Britain/Internet **[1]**

**(c)** Advantage:
-Each user has access to different types of printer for different jobs/more than one printer in case of failure/ease of maintenance
Disadvantage:
-Need to queue for printout/printouts not private/printer may not be local/No direct control over printer/lack of back-up if printer fails
(1 Advantage, 1 disadvantage) **[2]**

**3** **(a)** -Windows…
bounded areas of screen within which different tasks can be running

-Icons…
pictures/symbols/represent different options

-Menus/taskbar…
the listing of options from which the user may choose

-Pointers…
allows simple navigation/usually controlled by mouse (other pointing device)/used to select appropriate option.

-Highlighting…
altering appearance/dragging and dropping

(Any 3 x 2 points) **[6]**

**(b)** -PC is able to run more than one task…
-**apparently** simultaneously/at the same time
(1 for running more than one task at a time, 1 for apparently, max 2) **[2]**

**4** **(a)** -Management information
-Credit card verification
-To enable shared access to the database of seats/to allow each computer access to up to date information
-To avoid double booking
-To ensure that too many seats are not sold
-The information would be unreliable/inconsistent if held individually
(1 per -, max 2) **[2]**

**(b)** Input:
-Magnetic stripe reader/chip reader/card reader…
-to read data from credit card for payment
-Touch screen…
-to input details of purchase to a menu screen
Output:
-Printer…
-to print the ticket once bought.
Storage:
-Central hard drive/server…
-to hold details of seat availability/allow any alterations, quickly
Communication:
-Network cards…
-to connect computer to network/identify computer
-Cable/hub/switch/router…
-to connect computers/server.
(4 pairs of points, can only score second mark if first is awarded, max 8) **[8]**

**5** **(a)** **(i)** Data is sent one bit at a time/along a single line **[1]**

**(ii)** One byte is sent at a time/along the same number of channels as bits in a byte/multiple bits sent simultaneously **[1]**

**(iii)** Data can be transmitted in only one direction **[1]**

**(iv)** Data can be transmitted in both directions…
at the same time. **[2]**

**(b)** **(i)** -Number of ones in a byte is either always odd or always even
-System has been initialised to always expect either odd or even number of ones in a byte
-Each byte has one bit called the parity bit…
-which does not contain data…
-but is set to 1 or 0 in order to maintain the agreed odd or even parity
-When received, number of ones should match agreed parity or byte is in error
(1 per -, max 3)
**[3]**

**(ii)** -In the example odd parity is used
-01011010 is incorrect…
-because it contains an even number of ones.
(1 per -, max 2)
**[2]**

**6** **(a)** -ROM is not affected by turning off power/data on RAM is lost OR RAM is volatile ROM is not volatile
-Data stored on ROM cannot be altered/that on RAM can be changed
-ROM is smaller than RAM
(1 per -, max 2) **[2]**

**(b)** Hard drive:
-Storing operating system
-Storing applications software
-Storing user produced files
DVD:
-Importation of software
-Playing music
-Backing up files on hard drive
-Copying other DVDs
-Stores large permanent files eg, encyclopaedia
-Creating archive
-Transport data
(1 per -, max 2 and 2, max 4) **[4]**

**7** -Control unit
Manages execution of instructions/answer that makes it clear that CU controls rest of processor.

-Memory unit/IAS
Data/instructions are held (allowing access by the processor)

-Registers
Locations used for specific purposes

-ALU
Where data is processed/manipulated/(binary) arithmetic is done/I and O pass through

Note Three different specific registers named and described gets full marks.

(Any 3 x 2, max 6) **[6]**

**8** **(a)** **(i)** 01111100 **[1]**

**(ii)** 0001 0010 0100 (1 for 12 bits, 1 for answer) **[2]**

**(iii)** 174 (1 for 1, 1 for 74) **[2]**

**(iv)** 7C (1 per digit) **[2]**

**(b)** -Octal is binary digits taken in groups of 3
-from the right/add leading noughts at the left **[2]**

**9 (a) (i)** The characters that can be recognised by a computer/operating system **[1]**

**(ii)** –Each character is assigned a unique
-binary code…
-The number of bits necessary to code each character is a byte/2 bytes
-Commonly, ASCII codes are used
-Assigning 7 bits/8 bits to each character
-EBCDIC/extended ASCII/using 16 bits per character are alternatives.

(1 per -, max 3) **[3]**

**(b) (i)** -Whole number
-Counter (1 to 1000) for control of loop/identification of record

**(ii)** –Yes or no/True or false/2 state variable
-To change value when record is found/finish loop when record is found/label the correct record. **[4]**

**10 (a) (i)** 45

**(ii)** 66

**(iii)** 45 **[3]**

**(b) (i)** –Clash/more than one record gives same hash/01314 and 04505 give the same result. **[1]**

**(ii)** –Read addresses serially from hashed address…
until a space appears

-Use a bucket…
read duplicates serially in bucket until space is found

-Treat original as head of a linked list…
follow list to null value and add new value to end of list

(Any pair of points) **[2]**

**11** Comments/annotation
-Inserted as lines of code
-used to explain code
-not translated
-meaningful names used
-for subprograms
-and variables/objects
-in order to make code easier to follow
-self explanatory/does not require reference to table of names.
-Indentation of code
-Parts of code of a related type
-indented from left hand margin
-in order to show code lines that go together at a glance
-makes finding of beginning and end of loops and selection statements easy
-Modularity/top down approach
-dividing code into small sections
-easier to amend program
-easier to test and debug
-individual modules can be reused

(2 x 1 mark for name, 2 x 2 marks for description, max 6) **[6]**

**12** -Each terminal is (relatively) dumb
-Only one computer
-Each terminal allocated a time slice of…
-the single processor's time
-Each terminal has a time slice in turn
-When all have had a time slice the first terminal has another/round robin
-Allows terminals to have priorities/different slices/more frequent slices
-Use of flags to signify need of slice or otherwise
-Triggers that next terminal is moved to
-If slice not completed moves to next terminal.
(1 per -, max 6) **[6]**

**Total [86+4]=90**

# Mark Scheme 2507
# June 2005

**Task 1 [43 marks]**

**(a)** **(i)** Give 1 mark per attribute, the mark for each attribute may only be given if the data type is correct and there is a sensible description.

| Attribute | Data Type | Description |
|---|---|---|
| LecturerID (key) | Any allowable | Unique attribute to identify the lecturer |
| Name | String/Text/Char | The name of the lecturer |
| Office | String/Text/Char | The number of the office the lecturer uses |
| Phone | String/Text/Char | A four digit extension number |

Give 1 mark for identifying the key. **[5]**

**(ii)** Give 1 mark per attribute, the mark for each attribute may only be given if the data type is correct and there is a sensible description.

| Attribute | Data Type | Description |
|---|---|---|
| Module Code (key) | String/Text/Char | Unique attribute to identify the module |
| Name | String/Text/Char | The name of the module |

Give 1 mark for identifying the key. **[3]**

**(iii)** Give 1 mark per attribute, the mark for each attribute may only be given if the data type is correct and there is a sensible description.

| Attribute | Data Type | Description |
|---|---|---|
| Module Code (key) | Must match (ii) | Unique attribute to identify the module |
| LecturerID (key) | Must match (i) | Unique attribute to identify the lecturer |

Give 1 mark for identifying the two attributes as a composite key. **[3]**

**(iv)** Give 1 mark per attribute, the mark for each attribute may only be given if the data type is correct and there is a sensible description. (Other attributes may be given, but should be ignored for marking purposes.)

| Attribute | Data Type | Description |
|---|---|---|
| BookID(key) | String/Text/Char | Unique attribute to identify the book |
| Title | String/Text/Char | Title of the book |
| Author | String/Text/Char | Author(s) of the book |

Give 1 mark for identifying the key. **[4]**

**(v)** Give 1 mark per attribute, the mark for each attribute may only be given if the data type is correct and there is a sensible description.

| Attribute | Data Type | Description |
|---|---|---|
| BookID(key) | Must match (iv) | Unique attribute to identify the book |
| Module Code (key) | Must match (ii) | Unique attribute to identify the module |

Give 1 mark for identifying the two attributes as a composite key. **[3]**

**(b)** Give 1 mark for each answer. There must be proof that the candidate has created the validation checks.

**(i)** Mask
**(ii)** Mask
**(iii)** Mask (Do not accept integer/number/etc)
**(iv)** Drop down list or Use of referential integrity
**(v)** Drop down list or Use of referential integrity **[5]**

**(c)** Give 1 mark for each of the following to a maximum of 10

There are at least 5 modules
There are at least 10 books
There are at least 15 lecturers
Each LecturerID in the LecturerModule table is in the Lecturer table
Each Module Code in the LecturerModule table is in the Module table
Every Module Code is in the LecturerModule table
Every LecturerID is in the LecturerModule table
Each Module Code in the ModuleBook table is in the Module table
Each BookID in the ModuleBook table is in the Book table
Each BookID in the Book table is in the ModuleBook table
Each Module Code in the Module table is in the ModuleBook table
There is a module with only one lecturer
There is a module with more than one lecturer
There is a module with only one book
There is a module with more than one book **[10]**

**(d)** Give 1 mark for each of the following.

The report contains the full details of each lecturer
Evidence that the database software produced the report

Give 1 mark per point to a maximum of 3

The report has a sensible heading explaining the purpose of the report
The report has a date
The details are grouped by Module
The report contains the Module Codes **[5]**

**(e)** Give 1 mark for each of the following.

The report contains the full details of each book
Evidence that the database software produced the report

Give 1 mark per point to a maximum of 3

User can enter lecturerID
The report has a sensible heading explaining the purpose of the report
The report has a date
The report contains the lecturer code **[5]**

**Task 2 [25 marks]**

**(a)** **(i)** Allocate marks as follows

Give 1 mark if all the first three rows are correct
Give 1 mark if the Row values are all correct
Give 1 mark if the Col values are all correct
Give 1 mark if the Count values are all correct
Give 1 mark if the Number values are all correct
Give 1 mark if Array(1,1), Array(1,2), Array(1,3) are all correct
Give 1 mark if Array(2,1), Array(2,2), Array(2,3) are all correct
Give 1 mark if Array(3,1), Array(3,2), Array(3,3) are all correct
Give 1 mark if the NewRow values are all correct
Give 1 mark if the NewCol values are all correct

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **Size** | 3 | | | | | | | | | | | | | | | |
| **Start** | 1 | | | | | | | | | | | | | | | |
| **Step** | 1 | | | | | | | | | | | | | | | |
| **Row** | 1 | 3 | 2 | 3 | 2 | 1 | 2 | 1 | 3 | 1 | | | | | | |
| **Col** | 2 | 1 | 3 | 3 | 2 | 1 | 1 | 3 | 2 | 2 | | | | | | |
| **Count** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | | | | | | |
| **Number** | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | | | | | | |
| **Array(1,1)** | 6 | | | | | | | | | | | | | | | |
| **Array(1,2)** | 1 | | | | | | | | | | | | | | | |
| **Array(1,3)** | 8 | | | | | | | | | | | | | | | |
| **Array(2,1)** | 7 | | | | | | | | | | | | | | | |
| **Array(2,2)** | 5 | | | | | | | | | | | | | | | |
| **Array(2,3)** | 3 | | | | | | | | | | | | | | | |
| **Array(3,1)** | 2 | | | | | | | | | | | | | | | |
| **Array(3,2)** | 9 | | | | | | | | | | | | | | | |
| **Array(3,3)** | 4 | | | | | | | | | | | | | | | |
| **NewRow** | 0 | 3 | 2 | 1 | 3 | 2 | 1 | 0 | 3 | 2 | 1 | 0 | 3 | 2 | 4 | 1 |
| **NewCol** | 1 | 0 | 3 | 2 | 3 | 2 | 1 | 0 | 3 | 1 | 0 | 3 | 2 | 1 | 2 | |

**[10]**

**(ii)** Give 1 mark if all values are correct

| | | Col | | |
|---|---|---|---|---|
| | | 1 | 2 | 3 |
| Row | 1 | 6 | 1 | 8 |
| | 2 | 7 | 5 | 3 |
| | 3 | 2 | 9 | 4 |

**[1]**

**(b)** Give 1 mark for each correct row
Give 1 mark for each correct diagonal

| | | Col 1 | 2 | 3 | 4 | 5 | 6 | 7 |
|---|---|---|---|---|---|---|---|---|
| | 1 | 57 | 39 | 21 | 3 | 97 | 79 | 61 |
| | 2 | 59 | 55 | 37 | 19 | 15 | 95 | 77 |
| | 3 | 75 | 71 | 53 | 35 | 17 | 13 | 93 |
| Row | 4 | 91 | 73 | 69 | 51 | 33 | 29 | 11 |
| | 5 | 9 | 89 | 85 | 67 | 49 | 31 | 27 |
| | 6 | 25 | 7 | 87 | 83 | 65 | 47 | 43 |
| | 7 | 41 | 23 | 5 | 99 | 81 | 63 | 45 |

**[9]**

**(c)** **(i)** Give 1 mark for each point

An error would occur
Because the column index would not be an integer **[2]**

**(ii)** Give 1 mark for each point to a maximum of 3.

Accept an algorithm that does the following.

After input insert an instruction
To test that Size is an odd number
If not, produce an error message
And stop **[3]**

**Task 3 [31 marks]**

**(a)** Give 1 mark for setting up a 1-D array of 100 cells
Give 1 mark for initialising all the cells to zero **[2]**

**(b)** Give 1 mark if the code is fully annotated
Give 1 mark if meaningful names are used throughout (for variable names, command buttons, text boxes, etc.)

Give 1 mark per point to a maximum of 3.

- User can enter position and lengths of 10 snakes
- User can enter position and lengths of 10 ladders
- User can only enter positive integers
- All snakes lie completely on the board
- All ladders lie completely on the board
- The lengths of snakes and ladders can only be in the range 10 to 30 **[5]**

**(c)** Give 1 mark if the new code is fully annotated
Give 1 mark if meaningful names are used throughout the new code (for variable names, command buttons, text boxes, etc.)

Give 1 mark per point to a maximum of 6.

- Initial square is set to zero
- A random number 1 to 6 is generated for each move
- New position is calculated
- If new position is not on the board, a move is not made
- Number of throws is correctly calculated
- Loop terminates when player is on square 100
- Number of 'throws' is output **[8]**

**(d)** Give 1 mark if the new code is fully annotated
Give 1 mark if meaningful names are used throughout the new code (for variable names, command buttons, text boxes, etc.)

Give 1 mark per point to a maximum of 2.

- User can input the number of simulations
- Only positive integers are accepted
- Program does the correct number of simulations
- Average number of throws per game is correctly calculated
- The average number of throws per game is output **[4]**

**(e)** Give 1 mark if the new code is fully annotated.
Give 1 mark if meaningful names are used throughout the new code (for variable names, command buttons, text boxes, etc.).

Give 1 mark per point to a maximum of 4.

- User can input the number of snakes and the number of ladders
- The number of snakes and the number of ladders can only be in the range 5 to 10
- Both numbers can only be integers
- Solution stores details of snakes correctly
- Solution stores details of ladders correctly
- Program works correctly for given number of snakes and ladders **[6]**

**(f)** Give 1 mark per point to a maximum of 6.

The program correctly validates each of the following.
(Only give the mark if the code is clearly marked to show where the test is made **AND** there are **two** sets of data for the test, one valid and one invalid.)

- There is only one snake's head on a square
- Only one ladder starts on a square
- There is not a snake's head at the top of a ladder
- There is not a snake's head at the bottom of a ladder
- There is not a snake's tail at the top of a ladder
- There is not a snake's tail at the bottom of a ladder
- There is not a snake's tail and a snake's head on the same square
- There is not a ladder's top and a ladder's bottom on the same square
- Ladder does not go off the board
- Snake does not go off the board **[6]**

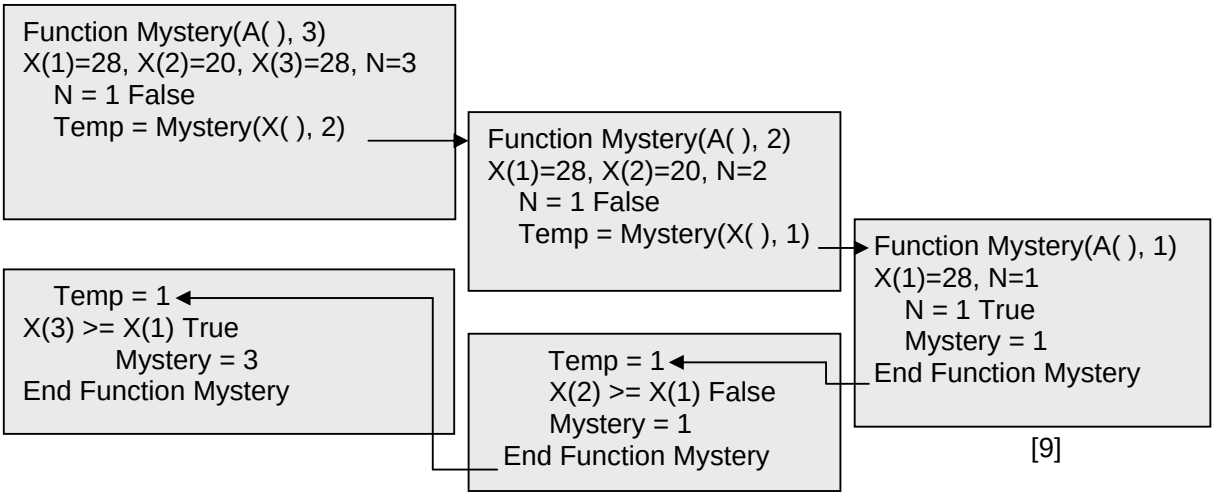
**Task 4 [21 marks]**

**(a)** Give 1 mark for each part only if the answer is exact. Numbers on their own are not enough.

**(i)** The answer is: 4 **[1]**

**(ii)** The answer is: 5 **[1]**

**(b)** Give 1 mark for each arrow and 1 mark for each shaded piece of code. Candidates who have written out the code in one block can receive the marks for the arrows providing the code blocks are in the right order.




[9]

**[9]**

**(c)** Give 1 mark per point to a maximum of 3.

- Returns the index
- Of the cell containing the largest value
- If two (or more) cells have the same highest value
- Returns the highest index **[3]**

**(d)** A typical solution is (but others are acceptable, see below for detailed mark scheme)

```
Function MysteryIterative(X( ), N)

    Index = 1
    Max = X(1)

    For Count = 2 To N
    If X(Count) >= Max Then
    Max = X(Count)
    Index = Count
    End If
    Next Count

    MysteryIterative = Index

End Function
```

Give 1 mark for each of the following points.

- Index initialised to 1
- Max initialised to X(1) (or other suitable value)
- Creation of a loop to look at the other values
- Test to see if a new Max is found
- If it is, reset Max
- And Index
- Returns correct value **[7]**

# Mark Scheme 2508
# June 2005

**1 (a)** Any six from:

- Problem Specification/identification
- Feasibility Study
- Analysis
- Design
- Coding/ software development/programming
- Testing
- Implementation/conversion
- Maintenance
- Review/evaluation

**[6]**

**(b) Problem Specification**
Existing problems identified (1)
User requirements (1)
As outlined by the warehouse (1)

**Feasibility Study**
Is it technically feasible? (1)
Software/hardware available (1)
Is it economically feasible? (1)
Is it socially feasible? (1)
Does the staff have the skills? (1)
Is it legal/ within data protection legislation?(1)
Is it ethically ok?(1)
Does it meet environmental standards? (1)

**Analysis**
Production of a requirements specification (1)
Detailing inputs needed (1)
and outputs required (1)
Specify user requirements (1)
Hardware/software needs (1)
Fact finding (questionnaire/interview/record inspection/observation etc) (1)

**Design**
Specification of system (1)
User Interface designed (1)
Specification of data structures (1)
Prototyping (1)
Algorithms (1)
Files/file structure (1)
Input/form design (1)
Output/report design (1)
Validation procedures (1)
Security of data (1)

**Coding/ software development**
Production of programs (1)
Tailoring software (1)
Modification of code (1)
Production of documentation (1)

**Testing**
Production and the carrying out of a suitable testing strategy (1)
Production of test data (1)
Alpha/beta testing/black/white box (1)
Different types of testing-extreme, normal, exceptional (1)
Specify expected results (1)

**Implementation/conversion**
Method of conversion/changeover (parallel, pilot, direct etc) (1)
Staff training (1)
Master file set-up (setting up data/transferring data) (1)
Installation of hardware (1)
Installation of software (1)

**Maintenance**
Debugging the system (1)
Modifications due to changing needs (1)
Improving current processes (1)

**Review**
Check against objectives stated (1)
Any limitations (1)
Any upgrades required (1)

**Any four of the above x 2 marks**

**[8]**

**(c)** **(i)** Any four from:

- Brief description of what the package does
- Description of input/output
- Machine configuration/hardware and software requirements
- Operating instructions
- Installation guide
- Simple error messages/troubleshooting/FAQs
- Tutorials
- On-line help/telephone help line
- Backup routines
- Glossary of terms

**[4]**

**(ii)** Any two from:

- To allow debugging of programs
- To assist future software developments/upgrades
- To modify existing file/data structures
- To support new technical staff

**[2]**

**2** **(a)** Any three from:

- **Pilot changeover**
- New system could be used
- In a few shops initially
- The results could be compared
- Against the other shops that use old system
- Roll out the new system
- If successful (abandon the old system).

**[3]**

**(b)** Any three from:

- **Direct changeover/ Big Bang**
- Old systems stop
- And the new system begins
- No overlap between systems
- No part changeover
- If new system fails
- Old system cannot be used

**[3]**

**3** **(a)**

- Automatic 3D generation (1) to show the different views of clothing item. (1)
- Libraries of design objects (1) to allow the fashion designer to include pre-drawn parts of clothing. (1)
- Create/edit/save/print or plot drawings (1) to allow further manipulation of the clothing items. (1)
- Interface with specialised hardware such as scanners/graphics tablets (1) to allow a previous drawing to be input. (1)
- Choosing fabrics/materials to construct the design (1) of various parts of the clothing. (1)
- Choosing colours/patterns (1) to develop different designs. (1)
- Calculating amount of material required (1) allowing for pricing of garment. (1)
- Rotate/zoom/scale features (1) to manipulate the finer detail of the garment. (1)
- Direct link to CAM (1) to realise/manufacture designs automatically. (1)

Any three of the above x 2 marks

[6]

**(b)** Any three from:

**Advantages of off-the-shelf**

- Readily available
- Third party user documentation available/on-line help/ support
- Shared development costs
- Tried and thoroughly tested/references from all users
- Extremely reliable
- Third party training
- Compatibility with other users of the software
- Regular updates/patches

Any three from: **[3]**

**Advantages of custom written**

- Designed to do exactly what the user needs/fits user requirements exactly.
- Programs can be written to run on existing hardware/comply with new hardware purchased
- Minimal change to working procedures of organisation
- User support can be geared to meet user needs
- No excess functionality
- Sell copies to other similar users to recoup development costs
- Program maintenance may be easier

[3]

**4** **(a)** Any two from:

- Sales of items can be identified quickly/stock movements can be monitored
- Low cost form of input
- High level of accuracy achieved
- Data capture is fast
- Eliminates human entry
- Items arrive from suppliers with bar codes on them
- Staff do not need a lot of training to use the bar code system
- Bar codes do not have to be aligned before reading
- Staff will feel confident in using such a system
- Prices of items can be easily changed

[2]

**(b)** Any two from:

- They are limited to fixed data
- Details cannot be changed
- If a bar code does not scan, manual entry can be slow/error prone
- Bar codes are easily damaged
- Some products do not have bar codes
- Cost of hardware may be expensive for smaller supermarkets

[2]

**(c)** Any three from:

- Encoded information stored on the strip can be automatically read into a computer
- Process of swiping is much quicker
- More accurate/not prone to human errors
- Information can be saved/transferred onto the strip
- Minimal training of staff required to use a magnetic strip reader (1)

**[3]**

**(d)**

- Range check (1)
  Check lower and upper limits of customer reference number. (1)

- Check digit. (1)
  Adding an extra digit to the customer reference number. (1)

- Existence check. (1)
  To check that a customer reference number appears/exists in the computer system. (1)

- Format/Character check. (1)
  To check all characters are of the correct type. (1)

- Field length check. (1)
  The number of characters entered are within the pre-defined limit. (1)

- Presence check. (1)
  To check that characters have been entered. (1)

Any three of the above x 2 marks

[6]

**5 (a)** Any three from:

- Requires the user to learn a large number of commands.
- Idea of prompting the user for a command.
- It also requires the ability to join commands to form instructions.
- The syntax in the command is critical. /will not operate if typed incorrectly.
- Not easy to use for the inexperienced user.
- Experienced users who are familiar with all the commands would find this interface quite fast compared to other types of interfaces.
- Single commands can allow the user to perform powerful operations

[3]

**(b)** Any three from:

- An on-screen form looks similar to a hard copy version.
- Similar layouts help transcription at the input stage.
- They enable data to be entered in a pre-determined and structured way.
- User normally enters data into boxes provided which makes it easier for novice users.
- Each box is labelled with a field name.
- List (or drop-down lists) boxes can be used to control what is entered.
- Radio buttons and check boxes can be used to enter data.
- If data types or data formats are not applied by the user automatic error dialogue boxes will appear.
- Help button available

**[3]**

**(c )** Any four from:

- Icons
- Pointers
- Menus/pull down menus/pop-up menus
- Windows
- Dialogue boxes
- Buttons
- Scrolling
- Toolbars
- Tool tips

**[4]**

**6** **(a)** Any two from:

- Document outputted by the computer.
- Information is added.
- and returned to the computer for further processing.
- Does not need to be manually inputted.

**[2]**

**(b)** Any four from:

- Batch processing.
- Meters read and data collected each quarter.
- Input as a batch, processed separately, outputted as a batch.
- Processed at a convenient time.
- Time of processing is not immediate.
- Normally at a quiet time such as middle of night/weekend.
- Similar data is processed.
- No human involvement during processing

**[4]**

**7 (a)**

| ***Data protection provision*** | *Applies to patient data* |
|---|---|
| • Personal data must be obtained and processed lawfully. | • Using patient details/consent of patient. |
| • Personal data must be held for specified purpose | • To assist the doctor in the well being of the patient. |
| • Personal data must be accurate and up-to-date. | • Such as recent prescription details. |
| • Personal data must be relevant. | • Only medical data held about patient. |
| • Personal data must not be kept longer than necessary. | • While the patient is still registered with the health centre. |
| • Personal data should be held securely, with no unauthorised access. | • From other non-medical staff in the health centre. |
| • Personal data should not be transferred outside the EU. | • Patient details should not be transferred to other medical professionals outside EU. |

**Max of 6 marks for "Data Protection provision" only**
**One mark for data protection and one mark for Patient data.**
**Example must be linked to Data Protection Act to achieve mark.**

[8]

**(b)**

- **User interface** (1)
  To allow communication between the expert system and the real world. (1)
  Doctors can input patient details. (1)
  Results can be fed back. (1)

- **An inference engine** (1)
  Contains the reasoning method used to search the knowledge base. (1)
  Looking for medical information. (1)

- **A knowledge base (1)**
  Contains facts and relevant data to a specified application. (1)
  Facts and relevant data about medical conditions. (1)
  Adding rules. (1)
  That can be applied to the facts. (1)

- **A rule base** (1)

Part of the knowledge base which is made up of all the rules. (1)
..known to the expert system. (1)

**Any three of the above: 1 mark for name of component and 1 mark for description in each case. Max of 6.**

[6]

**(c)** Any five from:

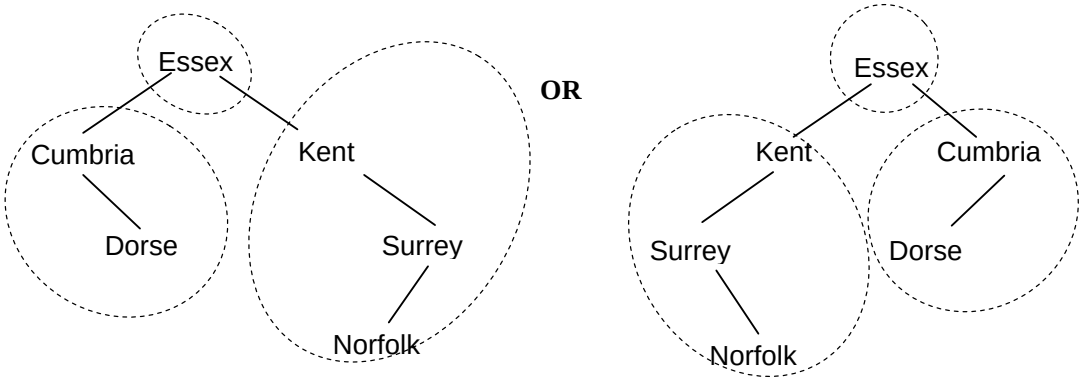
- Less time spent waiting to see the doctor.
- Patient needs to feel confident with the computer system.
- Only can be used for minor medical problems.
- May give the computer incorrect medical condition leading to an incorrect diagnosis.
- Patients with certain medical conditions may be able to spend more time with the doctor.
- Difficulty understanding the medical language.
- Accessibility/disability.
- Devaluing the role of the doctor as a medical expert.
- Doctor's workload may decrease/increase.
- Patient may leave without seeing the doctor/over reliance.
- Patient feels more comfortable with talking to doctor or a computer.
- Lack of human contact.
- Confidentiality – other patients may be able to see your medical details.

**[5]**

# Mark Scheme 2509
# June 2005

Each bullet point is worth **one** mark, up to the maximum for that section, unless stated otherwise.

**1** *Either diagram: 1 mark for root plus 1 for left subtree, 1 for right subtree.*




**[max 3]**

**2 (a)** ***Any three from:***

- Starts when PC switched on/reset
- Finds boot loader/on ROM (starts loading operating system)
- Mention of POST
- Initialises operating system
- Passes control to operating system/sets program counter
- Checks peripheral devices/checks hardware
- Checks memory
- Executed automatically
- Reads personal settings.

**[max 3]**

**(b) Paging**

- Used with virtual memory/allows programs to run despite insufficient memory
- Pages are stored on disk…
- Segments stored on disk (allow this only once)
- & assigned to memory when required
- Data/Memory is split into pages…
- …of equal size
- that do not need to be contiguous.

**Segmentation**

- Splits large program/data into smaller parts (called segments)
- Segments are different sizes
- Segments stored on disk (allow this only once)
- Each segment is a complete program…
- …that is executed separately
- Segments are run consecutively.

**(Max of 4 per type)**

**[max 6]**

**(c)** ***Any two points from:***

- Occurs when using virtual memory
- Very high rate of disk access
- More time spent loading pages than processing tasks
- Result is that program does not run/deadlock.

**Do not allow:** 'deadly embrace'

**[max 2]**

**3 (a) Interpreter**

- Translates & runs one instruction at a time
- Reports errors as they are detected
- Stops when an error is found
- Useful for program development
- Programs run slowly/programs translated each time they are run
- Allows program to restart from any point
- User can step through program
- Interpreter needs to be present in memory in order to run a program.

**Compiler**

- Translates whole program as a unit
- May report a number of errors together
- Many errors may be spurious
- Useful for program distribution (to retain copyright/prevent changes)
- Executable program runs quickly
- Produces object code/executable program/translation done once
- Compiler does not need to be present once code is translated.

**(Max 4 per type)**

**[max 6]**

**(b)**

- Lexical analysis (1)
- Syntax analysis (1)

**[max 2]**

**(c)** *Any **two** points from:*

- Produces a machine code program…
- … which is equivalent to the source program
- Several machine code instructions for each high level language instruction.

**[max 2]**

**(d)** *Any **two** points from:*

- Code is improved…
- … redundant code is removed …
- … to make the program run as fast as possible (runs efficiently)
- … to make the program use as little memory as possible
- Compiler decides whether to optimise for size or speed.

**[max 2]**

**4** **(a)** *Any **two** points from:*

- LIFO/FILO
- Pointer to top of stack
- Only item at top of stack can be accessed

(Diagram accepted)

**[max 2]**

**(b)** *Any **five** from:*

- Address of next instruction is copied from PC to MAR
- Increment PC (dependent on 1st mark)
- Instruction at that address is copied to MDR
- Copy instruction (from MDR) to CIR
- Decode instruction in CIR
- Execute instruction

**[max 5]**

**(c)** *Any **five** points from:*

- Current cycle completed
- Checks interrupt register
- Priorities of current task & highest priority interrupt compared
- That with higher priority secures processor attention
- If interrupt is highest priority
- Contents of registers put on stack
- Identify interrupt
- PC set to start location of interrupt service routine
- Interrupt processed
- Contents of registers pulled from stack
- Original task continues

**[max 5]**

**5** **(a)** *Any **two** points from:*

- Written in a standard format
- Avoid different codes meaning same value/to ensure each value is uniquely represented
- Form provides maximum precision
- More accurate multiplication

**[max 2]**

**(b)**

- 2nd example/ 01100 010
- change of bit in first 2 bits / starts 01

**[max 2]**

**(c)** *Any suitable calculation indicating out of range/overflow, e.g.:*

- 9 becomes 1001 in pure binary
- Mantissa 01001
- Exponent 4
- In two's complement 4 is 0100 / (3 bits, 100 represents –4)
- Exponent too large (so overflow)

or

- Maximum value is 01111 011
- Mantissa is 01111, exponent 011
- Represents 111.1
- So maximum is 7.5
- 9>7.5 so 9 can't be represented

(max 2 for calculation + 1 for reason)

**[max 3]**

**6 (a) (i)**

*Any **two** points from:*

- Also called imperative language
- 3rd generation language
- Tells the computer how to do something
- Gives a sequence of instructions
- e.g. BASIC / COBOL / FORTRAN / C / Pascal

**[max 2]**

**(ii)**

*Any **two** points from:*

- A variable defined within one part of program…
- & is only accessible in that part
- Data contained is lost when execution of that part of program is completed
- The same variable names can be used in different modules

**[max 2]**

**(iii)**

*Any **two** points from:*

- A variable that is defined at the start of a program…
- & exists throughout program…
- …including functions/procedures
- Allows data to be shared between modules

**[max 2]**

**(b)** *Any **two** points from:*

**(i) Class**

- A template…
- that define data and methods
- e.g. Product / Paint / Brush

**[max 2]**

*Any **two** points from:*

**(ii) Derived class**

- A more specialised class…
- that inherits the features of its superclass…
- & has additional attributes/methods
- e.g. Paint inherits ProductCode from Product (or similar from the example)

**[max 2]**

Any **two** points from:

**(iii) Object**

- An instance of a class
- e.g. a particular tin of red paint (any suitable example)
- *(accept* "A definition of data & permitted operations" - *BCS)*

**[max 2]**

*Any **two** points from:*

**(iv) Data encapsulation**

- The hiding of data
- Data can only be accessed by operations defined for the class
- A way of maintaining data integrity
- e.g. the price of any product can be changed/accessed using SetPrice:

**[max 2]**

**7** **(a)** *Marks in pairs: any 3 examples, with reasons*

- Combo box/dropdown list for central heating…
- … limited options/to save time/to avoid errors
- Automatic generation of new HouseReference
- …to avoid duplicate values being entered
- Add buttons to confirm/cancel changes to data…
- …so user can correct errors
- Add navigation buttons to move between records…
- …to make the system easier to use
- Add a title to the form…
- …to identify it
- Add instructions/help messages…
- …to help inexperienced users
- Improve layout…
- …for ease of data entry
- Use of font size/colour…
- …to improve readability
- price box could be formatted…
- …to display pound sign

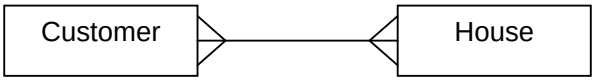
**[max 6]**

**(b)**

- Unique identifier/primary key

**[max 1]**

**(c)**




**(1 mark for each correct "many" end)**

**[max 2]**

**(d)** **(i)**


**(1 mark for each end of each relationship)**

**[max 4]**

**(ii)**

- CustomerReference is foreign key in Visit
- …and is primary key in Customer
- …provides the relationship between Customer & Visit

**or**

- HouseReference is foreign key in Visit
- …and is primary key in House
- …provides the relationship between House & Visit

**[max 3]**

**(e)** *Any **two** from:*

- To allow agency staff to access data easily (in a suitable format)
- To restrict access to confidential data (e.g. mortgage details)
- To protect data integrity/reduce risk of introducing errors
- Hiding complexity of data from the user/tailored interface for specific user

**[max 2]**

**8 (a) (i)**

- Going back to a previously found successful match

**[max 1]**

**(ii)**

- Giving a variable (in a statement) a value

**[max 1]**

**(b) (i)**

- rabbit(thumper).

**[max 1]**

**(ii)**

- likes(thumper, carrots).

**[max 1]**

**(iii)**

- no because food(carrots) is not in the rules

**[max 1]**

**(c) (i)**

- SUM calls itself within the algorithm

**[max 1]**

**(ii)**

- Recursive call is the last statement in the function

**[max 1]**

**(iii)**

```
PROD(N)
IF N = 1 THEN
PROD = 1
ELSE
PROD = N * PROD(N – 1)
ENDIF
END
```

- If N = 1 THEN
  PROD = 1
- ELSE
  PROD = N*PROD (N-1)

**[max 2]**

**Total 86(+4)=90**

# Mark Scheme 2511
# June 2005

**1 (a) (i)**

- A table containing members' personal details can be created using the database
- Data about members can be entered into this table
- These details can be kept up to date/modified/maintained
- A query can be used to select
- or filter out specific records
- mail merge

**[1] per point Max [2]** **[2]**

**(ii)**

- A standard letter can be created/retrieved using the word processor
- The letter can be edited/formatted as required
- The database/table of member details can be opened/attached to this letter
- Field markers can be inserted into the appropriate places in the letter
- The mail merge option can be used to print the personalised letters

**[1] per point Max [3]** **[3]**

**(b) (i)**

- Data/figures can be stored in the spreadsheet
- The data could be copied/extracted from other files
- Calculations can be performed/formulae can be used
- Data/results of calculations can be stored to a very high degree of accuracy
- Cells/figures can be formatted e.g. to display as currency
- Data/cells/formulae can be copied
- Macros can be created/used to perform regular tasks
- Graphics

**[1] per point Max [3]** **[3]**

**(ii)**

- Data can be transferred between the three components
- Relevant data from the financial records/accounts can be selected
- The data can be presented/summarised in the form of graphs
- … using a wizard
- … and default settings
- … or user-defined settings
- The graphs can be copied/linked to a word processed document
- … containing explanatory text
- macros

**[1] per point Max [3]** **[3]**

**2** **(a)**

- The network users will be able to share data/files/information
- … and software/programs
- … and peripherals/hardware e.g. disk drives/printers
- The network users will be able to communicate with one another
- Access to data/software/hardware can be controlled/central control
- Access can be monitored for accounting purposes/auditing
- Use any computer/station/terminal
- Data is up to date

**[1] per point Max [3]** **[3]**

**(b)**

- If the printer is ready, (the document sent for printing is sent directly to the printer)
- … under the control of the operating system
- (If the printer is not ready), the document is stored in a fast access device/wait/buffers
- … such as a disk drive
- References to documents are stored in a queue/in FIFO order
- (When the printer is ready), it prints the next job in the queue
- Spooling speeds up communication between two devices operating at different speeds/compensate for different speeds
- … such as the processor and the printer
- The creation of more than one queue if more than one printer is available
- Individual jobs may be given different priorities
- Frees processor for other tasks

**[1] per point Max [4]** **[4]**

**(c)**

- Each network user might be given specific access rights
- Groups of users might be given specific access rights
- A network user can be given access rights to a directory …
- … and/or access rights to specific files
- Access rights can be read only …
- read/write …
- no access
- Only some users may be able to read a file/the contents of a particular directory
- Only some users may be able to edit a file/the contents of a particular directory
- Access can be restricted to certain terminals
- Access can be restricted to particular times of the day
- The access rights are held in a table in memory
- … and checked with every access request
- … by using login & password
- files/directories/folder can be passworded

**[1] per point Max [3]** **[3]**

**(d)**

- A mailing list of all employees can be created
- Specific mailing lists can be created for different groups of users
- The same report can be sent to all users on a particular mailing list
- Attachments can be enclosed with the e-mail
- Replies can be sent back by simple menu choices/button clicks
- Attachments can be password protected/encrypted
- The sender can be notified automatically of the receipt/opening of an e-mail
- E-mails can be tagged as high priority
- The intended recipient can read the e-mail at any computer terminal
- E-mails can be sent at electronic speeds
- At any time
- Receiver does not have to be online
- Errors in emails reported to sender

**[1] per point Max [5]** **[5]**

**3 (a)**

- A Wide Area Network (WAN) (1)

- Network users over a wide geographical area
- … are connected by telephone lines
- … or satellite links
- routers/internet/gateway

**[1] per point Max [2]** **[3]**

**(b) (i)**

Distributed database system

- Several computers each hold copies of part of the data
- A database which is stored in more than one physical location

**[1] for one point** **[1]**

**(ii)**

Centralised database system

- A single computer holds the entire database
- The data is stored in a single physical location

**[1] for one point** **[1]**

**(iii)**

Distributed database system

- At regular intervals/overnight the copies of database must be reconcile **[1]**

Centralised database system

- As there is only one copy, the data is always up to date **[1]**

**(iv)**

Distributed database system

- The most common access to data will be local
- The priority will be to restrict/control local access
- … by using usernames/passwords
- Data must also be protected during transmission
- … by encryption/by error detection/correction techniques/parity bits

**[1] per point Max [2]**

Centralised database system

- The most common access to the data will be remote
- The priority will be to keep data transmissions secure
- … by encryption/by error detection/correction techniques/parity bits
- Remote access must be restricted
- Use of user names/passwords

**[1] per point Max [2]**
**[4]**

**4** **(a)**

- An entity model identifies the 'things'/actors/objects/tables/entities within a system …
- … in a specialised/standard diagram
- An entity model identifies the relationships/links between the entities …
- … such as one-to-one
- one-to-many
- many-to-many relationships
- An entity model identifies the attributes field of each entity
- ... the primary key attributes
- … and the foreign key attributes
- The result is a normalised model

**[1] per point Max [4]** **[4]**

**(b)**

- More productive/production can be fast
- Tools/wizards can be used
- Standard templates to start a model
- Components can be chosen from a databank
- Models can be edited easily
- Models can be reused
- More accurate/better quality models can be produced
- Only standard symbols will be used
- Automatic validation is performed
- The data dictionary can be populated automatically
- Documentation can be produced automatically
- Export
- Can be produced from data

**[1] per point Max [4]** **[4]**

**(c)**

- Project management tools
- These assist budgeting and resource allocation
- These assist the monitoring of the project
- These assist critical path analysis
- ... and the production of Gantt charts

- Code generators
- to produce code automatically …
- … from (module) specifications
- The code can then be edited if required

- Interface generators
- to produce code automatically …
- … from screen layouts
- The code can then be edited if required

**[1] per point Max [6]** **[6]**

**5 (a) (i)**

- The new system is implemented and tested in a single section/department/business area **[1]**

**(ii)**

- The organisation involved has a number of separate clubs
- … each performing the same function
- Any one of these clubs would be suitable for the pilot

- If the new system is satisfactory in one club
- … it can then be implemented in all the other clubs
- … with a high degree of confidence
- …can iron out errors

- The pilot can be used to train staff
- ... away from a live system

**[1] per point Max [4]** **[4]**

**(iii)**

- There is duplication of technical support for the current system and the pilot
- It is very difficult to test network aspects of the system from a single location
- Although successful in one location, the system may not be successful in another e.g. the data format used there may not be compatible

**[1] for one point** **[1]**

**(b)**

- The current IT skills level of the users
- Users with little experience will first require training in basic IT skills
- Users with experience can commence specific training in using the new system

- The most suitable format to use/the method to use
- Formal courses/training videos
- Some methods are rigorous, others are more flexible
- Different methods have different costs
- Cost of training

- The impact of change
- How similar are the new procedures to existing procedures?
- More change will require more training
- Time taken to train
- When training can take place

- The level of the user
- Day to day users will require training in at transaction level
- Managers will require training in using reports/information
- Numbers needing training

**[1] per point Max [5]** **[5]**

**6** **(a)**

- A web site contains a number of pages
- A page can contain text
- A page can contain tags/codes
- Tags may alter the text in some way e.g. font size
- Tags may be links to other pages
- Tags may be links to multimedia objects e.g. images
- A page is displayed on the user's screen by a browser
- The user navigates through the pages by clicking on links
- Pages are retrieved from the host site
- …and downloaded to the user's computer
- to ensure it works in a browser
- layout
- can include tables/images/video

**[1] per point Max [4]** **[4]**

**(b)**

- There is a potential global market with the web site e.g. people on holiday
- Any internet user could access the web site
- The content of the web site advertising could be kept up to date continuously
- The web site advertising could be multimedia
- ... interactive
- ... structured navigationally
- A search engine can be incorporated
- It can be customised to the user's preference/a history can be maintained
- Pop ups/intrusive advertising can be used
- The running costs of the web site advertising could be relatively low compared to e.g. distributing printed materials

**[1] per point Max [2]** **[2]**

**(c)**

- Members should be authorised/restricted in some way
- Digital signatures
- Each member should be given a username/password
- Additional credit card details should be requested
- … e.g. the 3 digit code on back of card
- After being entered for the first time, the full credit card details should not be displayed on the screen
- … e.g. leading digits should be replaced by *'s
- Data should be encrypted
- … prior to being transmitted over the internet
- Payments/transactions should be confirmed by automatic e-mails
- The clubs should have a clearly stated policy
- Firewall because
- …concern over hackers getting credit card details
- Secure connection
- …shown by padlock symbol

**[1] per point Max [4]** **[4]**

**7 (a)**

- A sensor could be attached to the volunteer
- … to record the heart beat
- … by detecting the electrical activity of the heart

**[1] per point Max [2]**

- A sensor could be attached to the volunteer
- … to record the rate of breathing/respiration
- … by detecting intakes of breadth

**[1] per point Max [2]**

- A sensor could be attached to the volunteer
- … to record how much the volunteer is sweating/losing fluid
- … by detecting moisture levels

**[1] per point Max [2]**

- A sensor could be attached to the volunteer
- … to record body temperature

**[1] per point Max [2]**

- A sensor could be attached to the volunteer
- …to record blood pressure

**[1] per point Max [2]**

- A sensor could be attached to the volunteer
- …to record oxygen levels

**[1] per point Max [2]**

**[2] for each of two sensors [4]**

**(b) (i)**

- A continuous/varying signal
- … such as the electrical signals from the heart
- … is sampled
- … and converted to digital form
- … for input into the computer/for storage on the computer system

**[1] per point Max [2] [2]**

**(ii)**

- A series/stream of digits
- … such as the stored representation of the heart beat
- … are converted to continuous/analogue signals
- … for displaying as a graph/on a monitor

**[1] per point Max [2] [2]**

**(c) (i)**

- An actuator is a piece of hardware/a device
- … which is controlled/reacts to signals from a computer
- … which produces a physical movement

**[1] per point Max [2] [2]**

**(ii)**

- Under control of a program
- … the computer sends a signal to an actuator
- … which controls the resistance to movement of the bike's pedal
- … making it harder to pedal

**[1] per point Max [2]** **[2]**

**(iii)**

- The system must respond immediately …
- ... to data being received about the volunteer
- … in an emergency/risky situation
- … because life is at risk
- 'Non real time' processing would be too slow
- so that display is up to date
- data is changing quickly

**[1] per point Max [2]** **[2]**

**Total [86 + 4] = 90**

# Report on the Units
# June 2005

## Chief Examiner's Report

All Modules worked well with some signs of improvement in performance in all Modules.

2509 and 2511 showed improvement that appeared to be due to the new layout. This made candidates focus on what was required. This meant that there were fewer answers that described things that were not asked for.

Again there was evidence that candidates' lack of communication skills was having a detrimental effect. Although one AS examiner thought there was some improvement in this area, others did not. Candidates must learn to clearly express their answers so that there is no chance of misunderstanding by the examiners. Poor communication is the primary cause of many failed computer systems.

Candidates still fail to make use of the marks given on the examination paper. Also, candidates should understand the meanings of the words 'state', 'describe' and 'explain'. This can be taught by using past papers and mark schemes.

The number of errors in the paperwork for 2508 and 2510 was unacceptable. The moderators had great difficulty keeping to the dates laid down for completion of moderation. In future, the paperwork will be returned to Centres for correction, this can lead to a delay in the production of the results for those Centres who do not return the documentation in time. Also, many Centres were late submitting the Coursework. This can be very serious as it leads to a suspicion of dishonesty in that the candidates are having more time to complete their work than is permitted.

## Principal Examiner Report

## 2506 Introductory Computer Systems, Communications and Software

### General Comments

The paper seemed to be fair to candidates. There were no reports of candidates experiencing time problems and no reports of questions that did not provide the expected responses. Different questions were attempted with different degrees of success by the candidature as would be expected, but there was no evidence of candidates being led astray by poor wording or ambiguities in the questions and all questions provided the full range of marks from 0 to the maximum for the question.

There is evidence that there are fewer candidates who are unprepared for the questions than there are in the January session. This is presumably because the candidates are that much more mature and they have a better understanding of the expectations of the paper. However, there is a small, but significant, number of candidates who simply should not be taking the exam. As has been reported before, it must be a very dispiriting experience for them and some responses are so poor that one has to wonder if they have gained anything from nominally following a Computing course. The numbers that fall into this category are falling but many Centres have one or two candidates that fall into this category. Rather more impressive than this group, but still finding difficulty with the assessment, is the group who still answer as GCSE candidates. This group has a reasonable understanding of the specification but is lacking in basic examination technique. Use of simple things, like the marks available for a part of a question so that the candidate is providing enough evidence and learning to read the stem of the question rather than just latching on to a key word, simple techniques that can help many candidates significantly improve their mark. Centres are advised to spend some of the time leading up to future sessions in practising such techniques with their candidates.

Many candidates performed extremely well and produced work which was a credit to themselves and to the teachers who had prepared them. Presentation was excellent with all scripts being legible and giving the impression that the candidates cared about the impression that they were giving. Consequently it was possible to give full credit to candidates for what they were able to say and the examiners did not feel as though they were being made to read into answers.

### Individual Questions

1. Both parts of the question were generally well answered. Many candidates sensibly gave examples of utility and applications software and it is pleasing to note that very few used proprietary brand names. Some candidates simply reworded the question for their definitions and others failed to earn credit because they were unable to express their thoughts clearly due to a lack of vocabulary, This was particularly prevalent in part (b) where a simple concept was quite difficult to put into words.

2. a) An apology is owed to candidates. There was not enough space given for the answer here which meant that the vast majority of candidates limited themselves to one answer instead of two. There was plenty of extra space and pages of blank lines for continuing answers at the back, so candidates should have been able to provide a full answer (as many did) but it is understandable that some candidates decided that they had filled the allotted lines and consequently they must have answered the question. Perhaps this is a prime example of exam technique failing many because they had not fulfilled the question requirements. There can be no excuse for the candidates who described the difference as being that a LAN was local while a WAN was wide. This was a common answer and suggests the GCSE type of understanding of the concepts that was mentioned above.

   b) Well answered, though “Large company” is not enough for a WAN as large companies will tend to have many LANs.

   c) This was well answered although many candidates talked about cost for both disadvantage and advantage despite it being stated in the question that cost should not be considered. Many candidates talked about saving space as an advantage. After much discussion it was decided not to accept this as an advantage because it was not an advantage to the *use* of

networked printers. The acceptable responses for this and all other questions in the paper are available in the published mark scheme.

3. This question was well answered by the majority of the candidates. The question worked well because it was intended to be aimed at the E grade candidate except for the final mark which was for understanding that the multi tasking meant that multiple tasks could only *apparently* be run at the same time. Few candidates scored the final mark, but most candidates were able to describe three features of the GUI.

4. a) Most candidates were able to talk about overbooking and many managed the second mark, particularly as it was available for 'double booking' as this is rather different from just over booking the number of seats.

   b) Most candidates scored very well, although there were a number of scripts offering keyboard and mouse despite these being in the question, others offered software like databases, still others talked about RAM and ROM, probably because they saw them in another question and there was a liberal sprinkling of tills and microphones. However, candidates who stuck to the generally accepted pattern of one each from input, output, storage and communication hardware scored full marks.

5. a) Some full and accurate answers but many others that weren't. Many candidates were unable to adequately describe the terms, though there was evidence that they did know them. There was far too much of 'data', 'text' and 'byte' rather than 'bit' being transmitted. This was intended to be the easy part of the question before getting into the parity, but, for many, this proved too difficult as they simply did not know the definitions.

   b) There were the normal answers of the type 'the last one is in error because it is negative', but this concept is quite difficult so it was imagined that many candidates would have difficulty. This is a higher order concept and provided a good differentiator on the paper with some candidates giving full and accurate explanations.

6. a) Well answered, though many were struggling for a second difference and reverted to describing the uses to which they can be put.

   b) Again well answered, though an unfortunate penchant for illegal downloads was illustrated by many candidates.

7. Generally well answered, although some entire Centres showed a lack of understanding of the topic. Centres are reminded that the whole specification for 2506 can be examined and sections should not be missed out. Some candidates tried to score four marks by splitting the ALU into two separate parts: the AU and the LU, and many of the weaker candidates could only think of one part of the processor, which tended to be the memory. Generally, this question was a good discriminator.

8. This was very Centre dependent. Candidates generally find these manipulation concepts very straightforward and can look upon the question as something of a banker. However, when all the candidates from a Centre fail to provide an answer for (iv) for instance, it is apparent that hexadecimal had not been covered. The majority of candidates scored very well and the suspicion is that those that did not score well did not really have the chance to.

9. a) The suspicion is that lack of the ability to express their thoughts was partly to blame for the poor marks given here. Examiners cannot read too much into answers because they may be reading in what was never intended and also the skill of being able to express these ideas is an integral part of the assessment. Though it would be wrong to be too negative in the report on this question as it was intended to test the more able candidates. Some candidates gave excellent responses.

   b) "Integer is a number" was seen far too frequently. It is accepted that this is not a mathematics exam, but 'integer' is a computing term.
   This question discriminated well across the mark range.

10. Some candidates failed to work out the correct answers in part (a). The origins of some of the answers were fathomable by the exam team, but others remain a mystery. While most were able to score 4 marks here, the final two marks proved beyond all but a few candidates, most who attempted the question simply wanted to alter the algorithm or change the ID numbers.

11. Many candidates believe that everything in the specification should be on the paper. It is at this point in the paper that they decide that a particular topic that they revised should be included if they have not used it yet. This would explain why a sizeable minority suggested alpha/beta testing and white/black box testing for their answers to this question. This is a failure of technique again in that the candidate's expectation has overcome their knowledge.
Most candidates could describe one technique and a large proportion were able to describe two, as per the question. However, there were few who could score all the marks for the description and marks of 2 and 4 out of 6 were common.

12. Once again, poor exam technique was seen, this time among the more able candidates. There were six marks, for which it was necessary to say six things. Very few managed to do this. The question was intended to be aimed at the A grade candidates but even so it was disappointing to see the poor level of responses to the concept of multi-user. Those that attempted the question were generally describing network systems or a stand alone computer which could be used by different people at different times of the day.

## Principal Moderator's Report

## 2507 Computing Structured Practical Tasks

**General comments**

Many Centres now deal with the administrative side of this specification accurately and within the Board's deadlines. Others do not do so and the moderation process becomes very time-consuming for both the moderator and staff at the Centres.

By the May 15 deadline, Centres should send to their moderator the appropriate copy of their MS1 mark sheet(s), together with the Centre Authentication Form and the Coursework Cover Sheets for all candidates. The moderator needs to see the breakdown of marks if a sample is to be requested. Where there are 11 or fewer candidates, all coursework should also be sent at this stage. Other Centres should retain their coursework until a sample is requested. The sample must then be sent promptly. If Centres with fewer than 20 candidates wish to send the work of all their candidates with their paperwork, they are welcome to do so if they feel it makes the process more efficient for them.

Before sending any paperwork, Centres should make three clerical checks: that marks have been correctly transferred from the coursework to the cover sheet; that these marks have been added correctly and transferred accurately to the summary grid on the front of the cover sheet; and that the total mark has been transferred correctly to the MS1, with the appropriate lozenges shaded in. Many Centres make some clerical errors. Some Centres make many clerical errors. All delay the moderation process and cause further work for examination officers and teaching staff in Centres.

Candidates should not submit work in ring binders, nor in plastic wallets. Work should be page numbered, placed in the correct order (often it is not) and hole-punched for a single treasury tag in the top left corner. The work is then best placed in a clearly labelled envelope folder. All work should be identified with the candidate's name and the Centre and candidate numbers. The cover sheet is not suitable for this identification as it does not remain with the work.

Annotation by markers remains the key to both good assessment and efficient and reliable moderation. In many cases Centres ensure that their marking is clear and the work annotated to show the mark points given, but there are still some where all the moderator has to work with is a mark for the section. There were a number of Centres this year where no breakdown of marks for the sub-tasks was given and little or no marking was shown on the coursework. Although an annotated copy of the mark scheme is helpful, the best practice approach is to show clearly, on the candidate's work, where and why each mark has been awarded. The more straightforward the moderation process, the less likely it is that Centre's marks will need to be changed. It may be necessary in future for work to be returned to Centres for clarification where insufficient marking information is provided.

**Comments on individual questions**

**Task 1**

This gave most candidates an opportunity to show their computing skills. It was intended for the E/D candidates and a high proportion of the entry scored well.

A number of candidates used a numeric data type for the telephone extension number in the Lecturer table in (a). Some Centres allowed this, despite a clear indication on the mark scheme that a text field was required; moderators did not. Weak and ineffectual descriptions of attributes remain commonplace.

The link entities caused some problems. Many candidates introduced an auto-number key when a combination of the two foreign keys (Lecturer key & Module key) would have been more appropriate.

The use of masks appeared to be well understood by some in (b), although a significant number failed to use them appropriately for validation. Some candidates used A and 9 (which would allow spaces, for example), rather than L and 0, and should have been penalised. Many candidates supplied some evidence of the use of referential integrity, but did not explain how it enabled them to validate the link entities. Such an explanation was essential for the award of these marks.

Many candidates created acceptable reports in (d) and (e), but it was not uncommon for the layout to be poor and the headings were often meaningless. For example *LecturerModule Report* does not adequately describe a report detailing the lecturers teaching each module. The date of the report (very important) was commonly omitted. A surprising number of candidates allowed the parameter in the second report to be input as a name rather than an ID, despite the instruction in the question, and were incorrectly given credit.

**Task 2**

The grids caused few problems and most of these were, as might have been predicted, due to carelessness. Row, Col, NewRow and NewCol appeared to throw up the most errors. A small number of candidates insisted that they knew better than the question paper in (a)(i) and, incorrectly, produced a grid larger than the one provided. Some Centres provided copies of the task, with the blank grids, for candidates to complete their solutions. This is an acceptable approach.

Although most candidates scored highly, many did not appear to understand how the magic squares were created. Consequently, although they could provide an answer (often badly worded) in (c)(i) to explain the problem with the input of 8,1,1 to the algorithm, they were unable to correctly answer (c)(ii). The algorithm only works for odd ordered squares, thus an error message should have been output. Modifying the algorithm to use it with an even ordered square does not work and should not have scored marks.

**Task 3**

This was intended to challenge the more able candidates and it did. Most were able to cope with the first two parts and some interesting interfaces were designed. As the task progressed, fewer and fewer candidates were able to solve the problems. A few candidates produced very good solutions. Some spent a great deal of time producing elaborate designs and more complex code than was required by the questions. It is important that those who set out on this path understand that such answers will not gain them extra credit.

A significant number of candidates did not approach the sub-tasks in the order set or at least did not present their answers in this way. In some cases it was very difficult to identify the different sections, especially where the marker provided no help either. A particular fault resulted in some candidates losing marks, because they adapted their answer to (b) to provide their answer to (e), and in doing so changed important parts of their original answer. Both needed to be present. Providing a separate answer to (e) was a simple solution!

Many candidates failed to properly annotate their code, although there is perhaps some sign of improvement in this area. It should be possible to write comparable code given the annotation alone. Care should be taken with the presentation of annotation; for many it is clearly thrown in as an afterthought. As in previous years, many candidates did not use meaningful variable names so that it was much more difficult to follow their solutions. Boards called N, snakes called S, snake heads called SH and buttons called Button3 are not acceptable given today's high-level languages. These features will continue to attract compulsory marks in future mark schemes. Some Centres will need to mark these aspects of programming questions much more rigorously.

Few candidates made a serious attempt at (f). Where they did, it was unusual to find both valid and invalid tests as required by the question. Many had actually input valid data (so that they could test their invalid data sets!) but failed to draw attention to this or to list what data had actually been used.

**Task 4**

Many candidates omitted the words 'The answer is:' from their solutions and a lot more omitted the colon. This meant that they should not have got either of the first two marks. Many Centres gave credit regardless; the moderators did not.

Part (b) was well answered by the more able candidates who produced diagrams similar to those given in past papers. Others seemed to have no awareness of the lessons to be learned from previous years. A fault of some, often ignored by their markers, was to draw arrows carelessly so that they left or entered a box of code at the wrong point. Such inaccuracy should be penalised.

Part (c) was usually answered well. Most could see that the algorithm searched for the highest number, but not all spotted that it was the array index that was returned. A good number of candidates stated that it returned the highest index where there were multiple highest contents.

In (d) many candidates could not produce an iterative solution that worked. As with Task 3, some candidates did not think about initialising variables and counters. Many forgot to output the answer!!

## Principal Moderator's Report

## 2508 Computing Structured Practical Tasks

**General Comments**

Performance in this exam varied greatly from Centre to Centre, with some excellent answers produced by candidates. It was pleasing to see the thorough way in which some Centres prepared their candidates for this exam. Apart from detailed knowledge of Computing and their applications a small number of candidates are demonstrating better examination techniques, such as reading the questions carefully and taking account of the marks allocated for each question.

There was evidence to indicate that candidates had sufficient time to complete the paper and there was enough spaces provided for their answers. The layout of the paper with structured questions continues to assist candidates in answering the questions.

I would advise all Centres to make use of past papers, mark schemes and examination reports as an integral part of exam preparation. The majority of the Specification is well understood by Centres, particularly section 5.3.1. Section 5.3.5 still continues to cause concern. Candidates continue to demonstrate a lack of knowledge in the area of expert systems. As the paper focuses on computer applications, candidates would be advised to refer to the stem of the question in their answers. For example, in question 7 many candidates understood the principles of the Data Protection Act but they could not apply their knowledge to a Health Centre.

**Comments on individual questions**

Question 1

The majority of candidates were able to achieve at least half of the available marks. In part (a) most candidates were able to name at least four stages of the systems life cycle. It was disappointing to see candidates referring to "documentation" as a separate stage. In part (b) too often answers were often a rewording of the stage name. For example, in the design stage description some candidates would state, "…involves designing the system.." which is not worth any marks. It was pleasing to see candidates understand the different sections of user documentation in part(c). In part(c)(ii) the majority of candidates only scored one mark out of the two available. Answers were often vague with the focus on "..the programmer correcting faults in programs..". Candidates would be advised to study other purposes of technical documentation.

Question 2

Most candidates were able to demonstrate a good knowledge of the methods of changeover, but too often they selected the wrong method for each application given. In part (a) the key phrase was "..a chain of stores.." Some candidates named the method as "phased" but went on to describe "pilot" thus gaining credit for their description. In part (b) some candidates used names such as "instant" or "immediate" as opposed to "direct" but were able to describe it fully thus achieving marks.

Question 3

Few candidates obtained full marks in part (a) of this question. Although most candidates were able to name 2 features, there was evidence that the descriptions were often a rewording of the feature. Candidates who were able to relate their description to designing a range of clothing were scoring higher marks. In part (b) the majority of candidates were able to achieve half of the marks available. Answers to this type of question are improving. For example in part (i) rather than saying, " ..off-the-shelf is cheaper than custom written

software..", candidates are now saying " ..off-the-shelf is cheaper than custom written software, due to the shared development costs.."

Question 4

This question was well answered by many candidates. In part (a), (b) and (c) answers were good with the majority of candidates scoring at least half marks. Although most candidates were able to suggest reasons for using bar code systems in supermarkets some candidates are still using phrases such as "..it's faster.." or "..it's more accurate.." without justifying their wording. In part (d) the better candidates were able to achieve full marks by naming and describing three validation methods used. Some candidates confused validation and verification methods in their answers such as double entry for a validation technique.

Question 5

This question was poorly answered by many candidates, particularly in part (a) and (b). A lot of candidates reworded the question in their answers such as "entering commands" for a command driven interface. Answers to part (c) were better with candidates demonstrating a good knowledge of GUI's. It must be stressed to candidates that WIMP by itself does not constitute four different features of a GUI, but when fully expanded and explained can be four different features and hence receive full marks.

Question 6

In part (a) the majority of candidates failed to score, demonstrating little or no knowledge of turnaround documents. In part (b) it was pleasing to see candidates identifying batch processing as the method and being able to use appropriate features to justify their answers.

Question 7

In part (a) most candidates were able to state several principles of the Data Protection Act but could not relate these to "patient data". It must be stressed that candidates should answer these questions in the context given, otherwise they cannot achieve full marks. In part (b) few candidates were able to fully name and describe three components of an expert system. The majority of candidates were able to name and describe one of the components of an expert system. Candidates need to recognise that a database is not the same as a knowledge base. In part (c) candidates included detailed answers but too often candidates kept repeating the same point achieving only 1 mark for their answer.

## Principal Examiner's Report

## 2509 Systems Software Mechanisms. Machine Architecture, Database Theory and Programming Paradigms

### General Comments

It was pleasing to see a number of excellent scripts where candidates had prepared for the exam and learnt the detail required at this level.

Two main areas of concern arose. Firstly, a number of candidates have very poor exam technique. They are advised to check the mark allocation for each part of questions and to look for key words such as "state" or "describe". It seems obvious, but many candidates could have improved their marks by reading the questions more carefully. A small minority gave a poor impression by producing a careless scrawl that proved, in places, almost impossible to read.

Secondly, candidates are advised to familiarise themselves with the whole specification. It was irritating to see criticisms of question 7(e) that the question was unclear or incorrect: "views of data" are clearly included in section 5.4.6 (3rd bullet point) in the specification. They are also expected to learn about different programming paradigms. It is likely that any practical programming experience will be limited, but it is sufficient to learn the key definitions and a few simple facts. It may be helpful if Centres could emphasise this.

### Comments on individual questions:

Q1 This was answered quite well, though a minority of candidates did not know the style of diagram for a binary tree. This is a basic technique that candidates must learn. A few tried to do an insertion sort.

Q2 (a) Most candidates gained marks.

(b) Some good answers were given. A few candidates confused segmentation with disk sectors.

It would have been nice to see a clearer understanding that with paging it is the physical memory that is divided (and the programs fit in) while in segmentation it is the program that is divided (and made to fit the memory).

(c) Few candidates gained both marks. There was some confusion between disk threshing and disk formatting. A number of answers mentioned the computer slowing down due to disk threshing.

Q3 (a) Many candidates failed to appreciate that an interpreter runs each instruction when it is translated. As in previous examinations, a significant number of candidates confused interpreters and compilers. Many candidates tried to compare the speed of interpreters with compilers rather than considering the speed of execution of the actual program.

(b) These were generally well answered, though weaker candidates said only that
(c) code generation "generates code".

(d) A few excellent, detailed answers were given, though many candidates could not answer this. There was a fundamental misunderstanding that white space

and comments are removed during optimisation, while clearly this cannot be the case as only object code is available by this stage.

Q4 The less able candidates failed to answer the question, only guessing answers based on the information given in (a) and (c).

(a) A stack is a basic concept that candidates should be able to describe. There was some confusion between stacks and queues. Looking ahead to (c), many wrote that a stack is used to store interrupts.

(b) Some candidates gave excellent answers and had clearly learnt the topic well. It was very disappointing to see answers of the form "fetch then decode then execute" – at this level, it is obvious that a 5-mark question requires technical detail. For those who had attempted to learn some detail, many wrote that data was transferred from the PC to the MAR, then to MDR and CIR. They did not appear to know that the MAR indicates the address from which a value is transferred to the MDR.

(c) Again, some candidates gave detailed answers. Others reiterated their answers to (a) and then attempted to "store interrupts in a stack". Many thought that whole jobs/tasks were put in the stack rather than the contents of registers.

Q5 (a) Some good answers were seen, but these were in the minority. Many candidates failed to realise that a standard format is required by the computer rather than by a user. Precision was rarely mentioned.

(b) Many candidates appeared to guess, with few giving a valid reason for their choice.

(c) A few candidates gave excellent answers, though these were in the minority. When learning about binary numbers, candidates should know how to write the maximum and minimum values (in binary) for a particular representation. It is then relatively easy to convert these values if required.

Q6 (a) In (i), the obvious guess was to write about a language "using procedures".

As in previous examinations, the local and global variables evoked responses related to one or more computers, or to local and wide area networks. It is disappointing to see that candidates are unable to give correct answers to a type of question that has been examined a number of times.

(b) Some excellent answers were seen. Disappointingly, a number of candidates had not studied this, or had not understood the basic concepts.

Q7 (a) Many of the better candidates gained all 6 marks here, but it seemed that a number of candidates either ignored the instructions or did not understand them. Some were unable to gain any marks – a surprise, as this was an easy question. Many answers included lists of additional attributes, some candidates thought of a single feature such as a button or combo box and tried to give 3 examples of its use, while a few thought of improvements but did not give reasons so only gained half marks.

(b) The majority gave correct answers.

(c)

(d) Many correct diagrams were drawn in (i), though a significant number of candidates were unable to resolve the many-many relationship from (c). In (ii),

many vague or incorrect answers were given even following correct diagrams in (i). Candidates must not expect to gain marks for a foreign key unless they state the table in which it applies. Some candidates ignored the Visit table, even after using it correctly in their diagrams, and tried to give a foreign key linking Customer and House directly.

(e) Similar questions have been asked in previous examinations, though few candidates appeared to have heard of views of data. The minimum that candidates should know is that views can provide easy access to data (particularly for users inexperienced in databases) while also protecting confidential data.

Q8 (a) A lot of guesswork was used here, though a significant number of candidates gave excellent definitions.

(b) This was intentionally easy and most gained at least 2 marks. On this occasion, (iii) was marked leniently, awarding a mark for answers such as “carrots are not defined”. In future, candidates may be expected to give more technical detail such as identify where a program fails or writing additional code.

(c) Some excellent answers were given. Tail recursion caused some difficulty, though many candidates indicated correctly that the recursive function call was at the end of the function.

## 2510 Computing Project

The project marking was very realistic this year with many Centres assessing work at the same level as the moderation team; there were very few examples of serious over-marking with few adjustments being recommended. The main issue raised by the moderation is lack of end user evidence from some Centres/candidates. It is vital that a systems analysis project should demonstrate the candidate's ability to investigate a problem, collect, collate and interpret evidence to design and produce a working solution suited to the end user's needs. If candidates are to score well then there must be evidence of the ongoing relationship with the end user throughout the project. In some cases there is no evidence that an end user has been consulted at any stage, yet in any real situation the end user would be involved at all stages of the development. Students need to provide hard copy evidence to support this aspect of the project. Typically, interview plans and notes, any existing paperwork or evidence of existing systems, evidence that the end user has been consulted over the type of solution, designs and development. It is crucial that the end user 'beta' tests the solution before the final development stage and then tests the final solution, there must be evidence of this.

One recurring comment from the moderation team referred to the lack of discussion of alternative solutions. It is important that the end user be presented with a reasoned discussion of the possible approaches to the problem in order to justify the chosen approach. These should be realistic alternative approaches, one of which could be a manual solution, another could be an alternative approach within the same software package, but it is not really appropriate to identify a piece of software which clearly could not perform the task, for example, a word processor to solve a stock control problem.

The syllabus states specifically that the technical documentation must be a stand-alone document. While this is desirable, the moderators have been giving credit if items which belong in the technical guide have been specifically referred to by candidates from within this section. It is also important to note the syllabus requirement that for full marks for the user guide there must be good on screen help. For web based projects, on screen help will form the major part of the user guide, it is unlikely that a hard copy user guide would be aimed at web site audience. There should be supplementary guides for the end user who will be charged with managing the system after the developer has 'left', this should explain how to install, update, backup, troubleshoot and generally manage the system.

There has been an interesting range of projects this year, whilst Access is still the most popular choice with students; we have seen more good quality web based and programmed solutions than in recent years. There have been some informal contacts regarding the possibility of students attempting to produce programmed solutions for a problem other than standard database ones using VBA This approach is possible and, if the students have the appropriate skills, should be encouraged. The issue often raised is the apparent close link between the mark scheme and a database solution, but this need not deter students from pursuing other approaches and OCR offer coursework consultancy which can provide advice on individual project ideas. With programmed solutions for other than the traditional database problem, identifying suitable end users is often an issue, as is web site development. If the problem being attempted does not necessarily have an obvious single end user then it is perfectly acceptable to use a focus group from the target demographic. These focus groups can be used to discuss the requirements, review the development and report on the success of the project, but students should avoid taking on this role themselves; there must be an outside influence on the development of the project. It is important students feel able to demonstrate their systems analysis skills effectively with a project that also demonstrates their system development skills effectively.

There was also an increase in clerical errors compared to previous years; these do slow the moderation process as moderators are forced to wait for amendments to be sent. Please check additions, transcription of marks to MS1 forms and that the Centre Coursework Authentication sheet and individual candidate coursework assessment sheets are included in the mailing to the moderator. If the centre has 11 or fewer candidates, then also send the work for all candidates directly to the moderator without waiting for a sample to be requested.

## Principal Examiner Report to Centres

## 2511 Integrated Information Systems

### General Comments

The new format of the paper worked well. Candidates' answers were much more focussed; there were far fewer general descriptions that never answered the question set.

However, the quality of written communication continues to decline. Usually examiners have been able to work out what a candidate means. This year that was not always the case. The result was that a number of candidates did not gain as many marks as they should have.

There was no evidence that any candidates had insufficient time to complete the questions or that any questions were inaccessible. Some questions had excellent answers from all candidates while others only received good answers from a few candidates. Hence the paper did discriminate between the candidates.

### Individual Questions

1 This question produced answers that were often difficult to understand. Most candidates gained over half marks on all parts, but some candidates produced very vague, general answers. Nearly all candidates stated that the database held details of members or that a query could be used to find those members who were over 18 years of age; few gave both answers. A few candidates thought that all members were over 18. Its use in a mail merge was rarely mentioned.

Answers to (a)(ii) were often too vague. The use of a template was rarely mentioned nor was the use of field markers. Nearly all candidates mentioned mail merge.

Few candidates could give three features of a spreadsheet that could be used in financial transactions. Currency formatting and the use of formulae were the most common acceptable answers. Accuracy, copying, importing and exporting data and the use of macros were not seen.

(b)(ii) produced very vague answers. The idea of creating graphs using the spreadsheet and exporting the results to a word processor was the most common answer. However, the use of the database to produce personalised letters and the word processor to add explanatory text were rarely mentioned.

2 (a) was well answered, most candidates gaining full marks. Part (b) was not well answered by many candidates. Few candidates realised that documents are stored on a fast access device and references to the documents are stored in a queue. Candidates often mentioned FIFO, buffers, the need to compensate for different speeds, freeing up the processor so that it can perform other tasks and priorities. Unfortunately, most candidates mentioned only one or two of these answers.
In part (c), nearly all candidates mentioned login and passwords but very few discussed access rights.

The features of email were generally well understood and most candidates gained 3 or 4 marks.

3 Nearly all candidates gained a mark for Wide Area Network (WAN) but descriptions were often poor. A number of candidates thought that a WAN was one that spread over a wide geographic area (1 mark) and that the PCs were linked by cables (0 marks).

The answers to 3(b) were either very good or very poor and were very Centre oriented. A distributed database consists of a central database with copies of different parts of it in different locations.

The answer 'A database held centrally' was not acceptable in answer to (b)(ii). Many candidates thought that the main database was updated at the same time as the distributed parts of the database. This defeats the object of a distributed database.

Security was well understood by most candidates.

4 Almost all candidates gained full marks for part (a) and at least 2 marks for part (b). Part (c) was not well answered. Gantt charts and critical path analysis (CPA) were mentioned but very rarely described. Project management tools, code generators and interface generators were very rarely mentioned.

5 Surprisingly, very few candidates knew the meaning of the term pilot. Also, few candidates could relate their answers to (a)(ii) to the scenario. The same applied to (a)(iii).

Part (b) produced some very good answers, but many candidates repeated some of their answers. For example, changing the size and colour of text is only worth one mark as it is about text formatting. Answers need to be clearly distinct.

Nearly all candidates appreciated that using the web site for advertising increased the market and that it is cheaper than traditional methods of advertising. Candidates also mentioned that a web site can be kept up-to-date and can use multimedia to enhance the advertising.

Encryption was the most common answer to part (c) and many mentioned the secure socket layer (SSL). However, they were not able to describe it. Many candidates simply mentioned SSL; this was not worth a mark as it only shows that the candidate can remember three letters, not what they stand for nor what they mean.

7 (a) was an example of where candidates did not make use of the mark given at the end of the question. This part of the question was worth 4 marks for two ways; implying 2 marks for each. Candidates usually mentioned heartbeat or pulse and temperature, but they did not explain that in each case a sensor attached to the body was required. A few did mention a thermometer for temperature and a facemask for checking oxygen levels or breathing rate.

Many candidates did not take any notice of the third paragraph in the stem, which made it clear that body measurements were required and they discussed measuring the speed of the bicycle for which no marks were awarded.

AD and DA conversion were well understood.

Part (c) produced some very vague answers. Although most candidates knew that an actuator is hardware controlled by a computer, there were a number who seemed to think that it is a sensor. The idea of controlling the resistance to pedalling was generally well understood as was the need for real-time processing. Many answers to part (c) were difficult to mark because of the poor quality of written communication.

**Advanced Subsidiary (3820) &**
**Advanced (7820) GCE Computing**
**June 2005 Assessment Session**

## Unit Threshold Marks

| Unit | | Maximum Mark | a | b | c | d | e | u |
|---|---|---|---|---|---|---|---|---|
| **2506** | Raw | 90 | 70 | 61 | 53 | 45 | 37 | 0 |
| | UMS | 90 | 72 | 62 | 54 | 45 | 36 | 0 |
| **2507** | Raw | 120 | 98 | 85 | 72 | 60 | 48 | 0 |
| | UMS | 120 | 96 | 84 | 72 | 60 | 48 | 0 |
| **2508** | Raw | 90 | 66 | 57 | 49 | 41 | 33 | 0 |
| | UMS | 90 | 72 | 62 | 54 | 45 | 36 | 0 |
| **2509** | Raw | 90 | 66 | 57 | 48 | 40 | 32 | 0 |
| | UMS | 90 | 72 | 62 | 54 | 45 | 36 | 0 |
| **2510** | Raw | 120 | 98 | 87 | 76 | 65 | 54 | 0 |
| | UMS | 120 | 96 | 84 | 72 | 60 | 48 | 0 |
| **2511** | Raw | 90 | 62 | 56 | 50 | 45 | 40 | 0 |
| | UMS | 90 | 72 | 62 | 54 | 45 | 36 | 0 |

## Specification Aggregation Results

Overall threshold marks in UMS (i.e. after conversion of raw marks to uniform marks)

| | Maximum Mark | A | B | C | D | E | U |
|---|---|---|---|---|---|---|---|
| **3820** | 300 | 240 | 210 | 180 | 150 | 120 | 0 |
| **7820** | 600 | 480 | 420 | 360 | 300 | 240 | 0 |

The cumulative percentage of candidates awarded each grade was as follows:

| | A | B | C | D | E | U | Total Number of Candidates |
|---|---|---|---|---|---|---|---|
| **3820** | 9.6 | 28.2 | 50.0 | 71.6 | 86.9 | 100 | 1092 |
| **7820** | 13.7 | 33.1 | 56.5 | 80.4 | 95.8 | 100 | 839 |

**OCR (Oxford Cambridge and RSA Examinations)**
**1 Hills Road**
**Cambridge**
**CB1 2EU**

## OCR Information Bureau

**(General Qualifications)**
Telephone: 01223 553998
Facsimile: 01223 552627
Email: helpdesk@ocr.org.uk

**www.ocr.org.uk**

For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored

**Oxford Cambridge and RSA Examinations**
**is a Company Limited by Guarantee**
**Registered in England**
**Registered Office; 1 Hills Road, Cambridge, CB1 2EU**
**Registered Company Number: 3484466**
**OCR is an exempt Charity**

**OCR (Oxford Cambridge and RSA Examinations)**
**Head office**
**Telephone: 01223 552552**
**Facsimile: 01223 552553**