UNIVERSITY OF CAMBRIDGE INTERNATIONAL EXAMINATIONS

GCE Advanced Level

MARK SCHEME for the November 2004 question paper

9691 COMPUTING

9691/03   Written Paper, maximum mark 90

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 Examiners were initially instructed to award marks. They do not indicate the details of the discussions that took place at an Examiners' meeting before marking began. Any substantial changes to the mark scheme that arose from these discussions will be recorded in the published *Report on the Examination*.

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 must be read in conjunction with the question papers and the *Report on the Examination*.

- CIE will not enter into discussion or correspondence in connection with these mark schemes.

Grade thresholds taken for Syllabus 9691 (Computing) in the November 20 examination.

	maximum mark available	minimum mark required for grade:		
		A	B	E
Component 3	90	60	54	30

The thresholds (minimum marks) for Grades C and D are normally set by dividing the mark range between the B and the E thresholds into three. For example, if the difference between the B and the E threshold is 24 marks, the C threshold is set 8 marks below the B threshold and the D threshold is set another 8 marks down. If dividing the interval by three results in a fraction of a mark, then the threshold is normally rounded down.

November 2004

Advanced Level

MARK SCHEME
MAXIMUM MARK: 90
SYLLABUS/COMPONENT: 9691/03 Computing Written Paper 3

1   **Reply**

allows user to compose a response which is
automatically sent to original correspondent by remembering address

Filing

storing the message for future use in
A selection of user designed files

Copying/forwarding

Making a copy of the message and
Forwarding it to another person by using their address

Multiple forwarding

Sending copy of message to many recipients by
Using address book
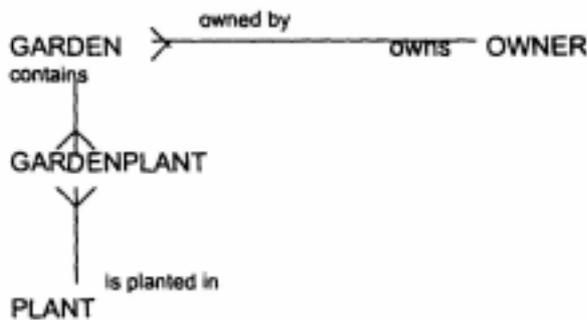
(Automatic) Deletion

Remove mail from message box after it has been sent to
Free up space

Blocking

If message is unsolicited and no further messages wanted from that source
Then provider will block future messages from that address

                              (2 per type, max 4 types, max 8)   **(8)**

2.

4 marks for entities
6 marks for relationships
4 for relationship descriptions

                              (1 per point, max 10)   **(10)**

3. **(i)** As jobs and files are loaded into memory they occupy space
which when vacated leaves gaps in memory
If a larger file is sent to that area and others
It has to be broken up to fit
This splitting of available memory into discontiguous pieces is called fragmentation

(1 per point, max 3) **(3)**

(ii) Memory is divided into
regular sized areas called pages
Jobs or files are allocated a number of pages according to size of job
Pages may be discontinuous
Index of pages/files kept
Addresses can be calculated by adding page address to raw address

(1 per point, max 3) **(3)**

(iii) Memory is divided into
variable length blocks called segments
Jobs or files can consist of many segments, different number each time taken
into memory
Segments normally match natural divide in jobs
Index of segments stored which must
Store base address and length of segment

(1 per point, max 3) **(3)**

(iv) A whole job does not need to be resident in memory at the same time
When a new page is required it is loaded into memory over a redundant page
from a reserved area of the hard drive
User believes whole job in memory simultaneously
Erased page may need to be saved first if it has been altered
Use of cache memory
Problem of threshing

(1 per point, max 3) **(3)**

4. Program is in modular form in machine code
The individual modules must be loaded by the loader into the computer's memory
Loader decides where modules are to be placed
Memory addresses adjusted by loader
Use of loader to load library routines
Linker links the modules correctly
Calculates addresses of the separate modules
Allows library routines to be linked to several programs

(1 per point, max 6) **(6)**

5.   (a)   Copy the address in the PC into
             the MAR
             Increment the PC
             Copy instruction at address held in MAR
             Into the MDR
             Copy contents of MDR into CIR
             Execute the instruction
             By altering the PC
             to the value in the address part of instruction
             Reset by returning to first step
             Mark for mention of check for interrupt

                                                 (1 per point, max 9)   **(9)**

      (b)   (i)   Many processors working together (on the same run of a program)   **(1)**

             (ii)   ADV speeds up processes because more than one calculation can be done
                    at a time
                    DIS Programs must be specially written   **(2)**

6. (a)   (i)   01101101   (1 for binary, 1 for 8 bits)
          (ii)   155   (1 for 1, 1 for 55)
          (iii) 6D   (1 for 6, 1 for D)   **(6)**

   (b)   (i)   01111111/01111111 =   127/128x2^127
                 01000000/10000000 =   1/2x2^-128 (2^-129)
                 10000000/01111111 =   -1x2^127
                 10111111/10000000 =   -65/128x2^-128   **(8)**

   (c)   Increase the number of bits used for the mantissa
          by reducing the number of bits for the exponent
          The range of numbers is reduced because
          The size of the index of the power of two is reduced

                                                 (1 per point, max 3)   **(3)**

7. (a)   (i)   A variable which is defined for a particular use in only one part of a program
                 /can only be used in block in which it is defined

          (ii)   Can be used and have its value manipulated throughout a program

          (iii) Information about a data item being supplied to a function or procedure
                 when it is called (and return data to calling routine)   **(3)**

(b) **A parameter passed by value**...
a local copy of the data is stored
in a separate location from the original
the value of the parameter can be manipulated
new value is discarded and return to original when procedure is terminated

A parameter passed by reference...
parameter is stored in original location
a pointer is passed indicating where parameter is stored
any changes will remove old value of parameter
new value available to calling program

(1 per point, max 5)   **(5)**

(c) **Empty stack**

100 entered as return address
6 as parameter
SP above 6
6 read by procedure
SP decremented
300,2,3 added to stack
SP above 3
3 and 2 read,
SP decremented twice
End of second procedure, return address read as 300
SP decremented
End of first procedure, return address 100 read
SP decremented

Accept any consistent use of SP

Diagrams showing the above are acceptable

(1 per point, max 6)   **(6)**

8. **(a)** Data collected about present conditions to include
wind speed/humidity/temperature/pressure
Also details about conditions close to required location
Predictions made (forecast) and
Prediction matched against actual results to
Hone predictions next time
Data collected by weather balloons/satellites/weather stations

(1 per point, max 5)   **(5)**

(b) Vast quantities of data
Large number of calculations
To be carried out in a very short time
Application is time sensitive
Application is processor bound
Parallel processing can increase processing speeds massively

(1 per point max 3)   **(3)**

(c) No travel involved
Documents/maps can be shown
Discussions can be held in real time
Saves time in an application where time is of the essence
Many people may be involved giving rise to discussions

(1 per point, max 3) **(3)**

Total (90)