CAMBRIDGE INTERNATIONAL EXAMINATIONS

GCE Advanced Level

MARK SCHEME for the October/November 2012 series

9691 COMPUTING

9691/31   Paper 3 (Written Paper), maximum raw mark 90

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge will not enter into discussions about these mark schemes.

Cambridge is publishing the mark schemes for the October/November 2012 series for most IGCSE, GCE Advanced Level and Advanced Subsidiary Level components and some Ordinary Level components.

CAMBRIDGE
International Examinations

1 (a) (i) -An attribute/or combination of attributes   A field
-Which (links to/is) the <u>primary key in a second / another table</u>   [2]

(ii) -the primary key from Entity X
-Matches to the foreign key
-The same key in table Y   [3]

(b) (i) -Two sensible attributes for Customer (but none which relate to the Product or Order tables)
-Two sensible attributes for Product (but none which relate to the Customer or Order tables)   [2]

(ii) -Order attributes include – CustomerID + ProductID
-Primary key of <u>CustomerID + OrderDate</u>   [2]

(c) *Data duplication*
-the same data is (unnecessarily) repeated in a second table
-Duplication means that data will be redundant in one of the tables
-A consequence of duplication is that data can become inconsistent   [2]

[Total: 11]

2 (a) (i) -108   [1]

(ii) 94   [1]

(b) (i) 7   [1]

(ii) Mantissa showing as:
½ + 1/16 + 1/64 // 37/64 // 0.578125   [1]
 +74   [1]

(c) (i) 0100 0010   [1]
0101   [1]

(ii) Normalised form the format for mantissa and exponent which ensures the maximum possible accuracy   [1]

[Total: 8]

3 (a) -single processor
-program consists of a sequence of stored instructions
-instructions + data
-are stored in a continuous block of main memory
-instructions are executed in sequence   [2]

(b) (i) -The contents of the <u>Program Counter</u> are incremented [1]

(ii) -Contents of the address held in MAR are copied to the MDR [1]

(iii) -the contents of the CIR are decoded
- the instruction is executed [2]

(c) (i) *Direct addressing*
The operand ....
is the actual memory address to be used

e.g. LD 1987 means copy the contents of address 1987 to the Accumulator register
(scores full 2 by example) [2]

(ii) *Indirect addressing*
-the operand part of the instruction is an address [1]
-this address contains the contents which are used [1]

(iii) *Indexed addressing*
-The processor will have an index register
-The contents of the index register are added
-To the operand (address) [2]

[Total: 12]

4 (a) *Key features of an interpreter*
-analyse the program statement by statement
-if the statement is valid then it is executed
-the interpreter will call program routines to execute each statement
-program executes until an error is found
-no object/executable code is generated
-the interpreter has to be run every time the program is executed [3]

(b) *Advantages of a compiler ...*
-The program will execute faster
-Execution does not require the presence of any translator software
-Once compiled the process allows for easy distribution of the executable file(s)
-Difficult to reverse engineer [2]

(c) *Lexical analysis*
-Keywords and identifiers (in the source program) are tokenised
-Unwanted characters e.g. <Space> and comment statement text are removed
-All keywords are matched to a dictionary/table of keywords
-Check for valid identifier names
-Against the (say BNF) rules for valid names
-Errors for either invalid keywords/identifiers are reported [3]

(d) *Code optimisation*

Possible process which follows the lexical analysis / syntax analysis/ code generation stages
Code produced by the code generation process may not be the most efficient code
Produce code which executes faster (than that produced by the translator software)
Produce code which takes up less memory when executed / reduces the amount of program code [2]

[Total: 10]

5 (a) Last item added to the stack will be the first item to leave [1]
NE LIFO

(b)
```
PROCEDURE PushToStack
    IF TopOfStack = 100                              [1 + 1]
      THEN
          OUTPUT "Stack is already FULL"
    ELSE
      INPUT NewItem
      TopOfStack  ← TopOfStack + 1                   [1]
      MyStack[TopOfStack] = NewItem                  [1 + 1]
    ENDIF
END PROCEDURE                                        [4]
```


(c) *Application of a stack*

-Any valid application e.g.
  o For the conversion of infix expression to reverse Polish
  o Interrupt handling
  o For the storage and retrieval of return addresses for procedure calling [1]

-Two marks for a clear explanation
  o Every time a new call is made
  o The return address must be stored
  o Return addresses are recalled in the order 'last one stored will be the first to be recalled' [2]

[Total: 8]

6 **(a)** **(i)** The job is removed from memory

This space released becomes available
OR …
move all current jobs in memory
Move all jobs loaded so that only one hole is present [2]

(ii) Look at the available 'holes' which exist
Allocate the job from the scheduled list ....
which occupies the largest available space in memory
OR …
Find the largest job ....
which can be allocated into the available space [2]

(b) **(i)** -a signal from some device
-to indicate that some event has occurred
-the device is seeking the attention of the processor [2]

(ii) *Any two ...*
-A peripheral e.g. printer
-to inform the processor it is out of paper/paper jam/ or similar

-user
-has pressed the 'Reset' button

-keyboard
-has generated an interrupt to say data has been entered and requires saving

-mouse
-has generated a signal e.g. click which will result in some action e.g. a refresh of the screen

-clock interrupt
-must complete the current f-e cycle

-software generated interrupt
-divide by zero error [4]

(iii) -Determine the source of the interrupt
-Mask out/disable all interrupts of a lower priority
-Save the contents of the Program Counter
-Save the contents of all other registers (on the stack)
-Load the appropriate Interrupt Service Routine (ISR)
-Run the ISR code
-Restore the contents of the registers
-Restore the contents of the PC
-Enable all lower priority interrupts
-Resume the next process [5]

[Total: 15]

7 (a) *Two different media*

-Copper wire
- o Use of the existing telephone network
- o Dedicated leased-line
- o Broadband enabled line

-Optic fibre cabling
- o Uses light // Data travels at the speed of light
- o High bandwidth possible
- o Many signals can be sent on a single fibre
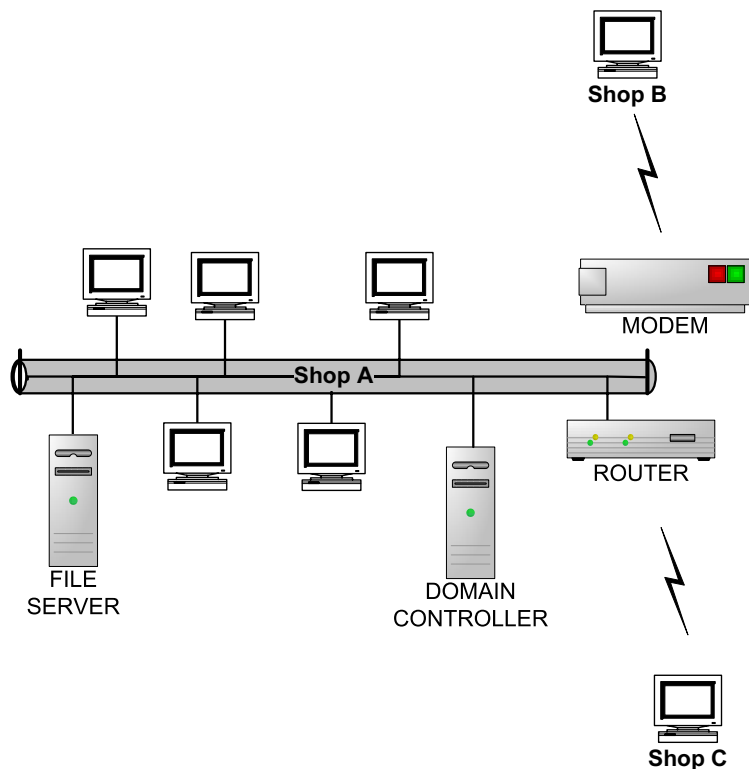
-Radio signals
- o Satellite communications

-Microwave signals
- o Communication must be a straight line from sender to receiver   [4]

(b)

Connections from Shop B and Shop C to Shop A
Server X (File Server) connected to the LAN
Server Y (Domain Controller) connected to the LAN
Modem (either at A, B or C)
Firewall (either at A, B or C)
Router (At Shop A only)   [4]

(c) Bridge
Allows for communication between the two segments   [2]

(d) (i) *Benefits of an Intranet*
Improved communication between shops
Provide information which is only available to company employees / restricted access
Limited access
Webmaster need only be concerned about the browser used by the company
Better security over company data/information [2]

(ii) Web server [1]

8 (a) (i) -function name ChangeString
-ThisString1 and ThisString2 [2]

(ii) Lyons W [1]

(iii) SMITH 9 [1]

(iv) Error [1]

(b) -self-contained blocks of code
-subprograms
-must be given an identifier
-may have parameters [2]

[Total: 7]

9 (a) Procedural language
Uses sub-programs/subroutines/ blocks of code
Procedures are self-contained
Program statements will be executed sequentially [2]

(b) (i) car(zx6) [1]

combination(gearbox2, a3) [1]

supplier_part(motorB, dealerD) [1]

(ii) the part will be guaranteed if that part's supplier is dealerD [1]

[Total: 6]