CAMBRIDGE INTERNATIONAL EXAMINATIONS

GCE Advanced Subsidiary Level and GCE Advanced Level

MARK SCHEME for the May/June 2013 series

9691 COMPUTING

9691/12 Paper 1 (Written Paper), maximum raw mark 75

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 May/June 2013 series for most IGCSE, GCE Advanced Level and Advanced Subsidiary Level components and some Ordinary Level components.

CAMBRIDGE
International Examinations

1 (a) buffer – any **one** from:
temporary storage area used to hold data before being transferred
allows for difference in working speeds (of processors and peripheral devices)

interrupt – any **one** from:
signal sent to the processor/CPU (which causes break in the execution of current routine) [2]

(b) (i) Any **three** points from:

data is transferred from (primary) memory to printer buffer
when the buffer is full, the processor can carry on with other tasks
printer buffer is emptied to printer
when printer buffer is empty, printer sends an interrupt to the processor requesting more data to be sent
according to priorities [3]

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

first (block) of data sent to the *first* buffer
whilst this data is being printed by the printer
next block of data is sent to the *second* buffer
when the *first* buffer is empty
data from the *second* buffer is then printed
meanwhile more data is then sent to the *first* buffer
this continues until all data has been processed by the printer [2]

2 (a) 1 mark for naming type of media + 1 mark for description/examples

magnetic media
surface coated with magnetic material
magnetic properties altered to represent 1s and 0s
used by hard disks, magnetic tapes, floppy disks

optical media
surface coated with light sensitive material
read/written by lasers
CDs use one spiral track
used by DVD-RAM, CD-R, CDROM, CDRW, blu-ray disc

solid state media
uses millions of tiny transistors
where movement of electrons controlled within a microchip
has no moving parts
used by memory sticks, MP3 players, cameras/mobile phones [6]

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

RAM
contents can be altered/written to
holds data/program currently in use
volatile memory/temporary memory/contents lost when switched off
usually has a greater memory capacity than ROM

ROM
contents can be read only/can't be altered
holds bootstrap/BIOS/system data
non-volatile memory/permanent memory/retains contents when switched off [2]

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

needs RAM to store instructions given by the user
needs RAM to temporarily store program controlling car
needs RAM to store current radio frequencies to control car

needs ROM to store the factory settings/basic instructions
needs ROM to store radio frequencies (etc.) understood by remote controller
needs ROM to store start up routines when car switched on [2]

3 (a) (i) Any **one** point from:

transmission is sent in one direction only
along a single data line
Reject single CABLE [1]

(ii) Any **one** point from:

transmission can be in both directions at same time
along several data lines/one data line per bit [1]

(b) (i)

letter	bytes adjusted for even parity							
C	**1**	*1*	*0*	*0*	*0*	*0*	*1*	*1*
O	**1**	*1*	*0*	*0*	*1*	*1*	*1*	*1*
M	**0**	*1*	*0*	*0*	*1*	*1*	*0*	*1*
P	**0**	*1*	*0*	*1*	*0*	*0*	*0*	*0*
U	**0**	*1*	*0*	*1*	*0*	*1*	*0*	*1*
T	**1**	*1*	*0*	*1*	*1*	*0*	*0*	*0*
I	**1**	*1*	*0*	*0*	*1*	*0*	*0*	*1*
N	**0**	*1*	*0*	*0*	*1*	*1*	*1*	*0*
G	**0**	*1*	*0*	*0*	*0*	*1*	*1*	*1*

(–1 mark for each error in the first column) [2]

(ii) 0 1 0 1 1 1 0 0 [1]

(iii) Any **three** points from:

character “P” flagged as having *odd parity* (row 4 in diagram)
parity byte sent with data i.e. 0 1 0 1 1 1 0 0
column 5 also has *odd parity* (or equivalent)
faulty bit must be in row 4 and column 5
idea of auto correction of fault (in row 4, column 5)

(Check if diagram has been annotated to show faulty bit) [3]

4 (a) 1 mark for name of device + 1 mark for reason

scanner
to produce an electronic/digital map version of the passport photograph
(scans) into computer readable format

digital camera/video camera
to produce an electronic image of the passenger's face
produces a similar format to the scanned image [4]

(b) (i) 1 mark for each point

use of a pressure sensor
sends data back to computer system
need for conversion to digital form (ADC)
computer calculates weight based on sensor data
this calculated value is compared / (or equivalent) to stored values
sends weight back to a small screen at check-in
if weight exceeds airline limit, operator warned at check-in by, e.g., a flashing screen or beeping sound / if weight below limit luggage accepted [3]

(ii) 1 mark per point

(labels printed in form of) a barcode
barcode is unique
use of barcode reader/scanner
barcode is used as a key field in passenger record
barcode read at each stage …..
... and this data is stored in passenger record
thus allowing tracking/whereabouts of luggage at any stage [2]

5 (a) 1 mark for naming type of diagram + 2 marks for description/examples

data flow diagram (DFD)
graphical representation of flow of data through the system
shows what data is input and output from the system
shows where data comes from and goes to and where it is stored
contains process, flow, store and terminators

systems flow chart/diagram
way of displaying how decisions are made to control events
shows devices (e.g. disk drives)
shows media use for input/output and storage (e.g. paper output)
shows what files are used in the system

Program flowchart/Jackson structured diagram
shows method of solution
shows data inputs and outputs
shows modularisation

If NO valid diagram stated then cannot get features for that part of the answer
REJECT flowchart alone
(accept examples/diagrams as part of explanation reflecting description above) [3]

(b) 1 mark for each item + 1 mark for reason

purpose of the system
this allows another programmer to know what the aim of the software is

program listing/coding
if an error occurs in the original program this allows another programmer to alter the coding to cure the fault

flowchart/algorithm
these will show the logical steps in the program or system so another programmer can follow the logic

input/output formats
allows another programmer to test the system to see if it produces expected results

hardware and software requirements
useful if a system needs to be upgraded so another analyst can decide exactly what hardware or software would need to be changed/ ensure new hardware is compatible

list of variables/data dictionary
this allows another programmer to understand what the variables represent so they can understand the coding

file structures
this is important so that another programmer knows if they are fixed length etc. and how the data is arranged in the file

test plans/sample runs/test data
this is important so that another programmer can see how the original software performed and to see how it was tested

validation rules
knowledge of these rules will let a programmer know what error trapping exists within the software
so that new data follows the same rules

meaning of error messages
this helps another programmer understand what is going wrong with the software [4]

6 (a) (i) 1 mark for each correct output device + 1 mark for each reason

production

dot matrix printer
factory environment could be dirty/dusty/damp
need to produce stickers from continuous stationery

design office

large computer screen/monitor
need to be able to see new designs very clearly
if screens are CRT it is possible to use light pens

(graph) plotters
to produce "blue-prints" which can be used to build prototypes
often necessary to produce full size drawings of new product

3D printers
these allow production of solid prototypes/models that work
reduces cost of tooling up to make a real working example

marketing

laser printers
producing large numbers of brochures/catalogues/flyers
quiet operation in office environment
high speed operation when producing large print runs

screens
to show dialogue/script when answering customer queries

headphones
to hear customer queries

finance

dot matrix/impact printer
pay slips produced on continuous stationery
"hidden information" in ready sealed envelopes

laser/inkjet printer
produces payslip for later sealing
none secure
NE to produce hard copy of a wage slip [8]

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

large coloured graphics showing process
green colour to show ON and red colour to show OFF
flashing/highlighted colours used to indicate error/attention required
layout needs to allow links to other screens showing other part of process
layout should be clear and easy to use
ability to "click on"/choose graphic to show status of the item
use of input devices such as touch screens, trackerballs and keyboards/keypads to allow selection etc.
use of output devices such as large screens, printers and speakers/ beepers to produce plant data and warn of problems [3]

7 (a) (i) Any **two** points from:
(use of a 2-dimensional array) **`Grid`** [1:3, 1:3]
use 1 value for "X" and 0 value for "O" or characters "X" and "O"
each square in game corresponds to position in array …..
... e.g. 1 value in **`Grid`** [1, 1], 0 value in **`Grid`** [2,1] and so on
each time a square is "filled in" the position is identified and appropriate value stored in correct position in array **`Grid`** [2]

(ii) integer/char [1]

(iii) 1/null/space [1]

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

each array position is scanned
each row is checked e.g. first row: **`Grid`** [1, 1], **`Grid`** [2, 1] then **`Grid`** [3, 1] and so on
computer checks to see if all three values in the row are the same …
… if they are all 1s then "X" has won; if all 0s then "O" has won *
each column is checked e.g. first column: **`Grid`** [1 ,1], **`Grid`** [1, 2] then **`Grid`** [1, 3] and so on
computer checks to see if all three values in the column are the same
… if they are all 1s then "X" has won; if all 0s then "O" has won *
each diagonal is checked e.g. **`Grid`** [1, 1], **`Grid`** [2, 2] then **`Grid`** [3,3] OR **`Grid`** [3, 1], **`Grid`** [2, 2] then **`Grid`** [1, 3]
computer checks to see if all three values in diagonal are the same …
… if they are all 1s then "X" has won; if all 0s then "O" has won *
this scan is carried out each time a square has been filled in
* this answer can only be accepted once

If an algorithmic approach has been applied mark along the following lines:

initialise flag before checking array
concept of looping for a row
concept of looping for a column
recognising position in an array
check symbol in adjacent cells is the same
flag or recognition of change in content [4]

(c) 1 mark for device + 1 mark for reason

touch screen
user only has to touch a square on screen to make choice
computer knows player is "X" and puts an "X" where screen touched

mouse/trackerball/trackpad
user moves cursor/arrow and points to chosen square
"click" to confirm and "X" is placed in chosen square

keyboard
move cursor/arrow by using ↑ ↓ ← → keys
press <ENTER> key when chosen square found [2]

8 **(a)** record key is used
the key value is subjected to arithmetic algorithm
to give/calculate the location/address of the record [3]

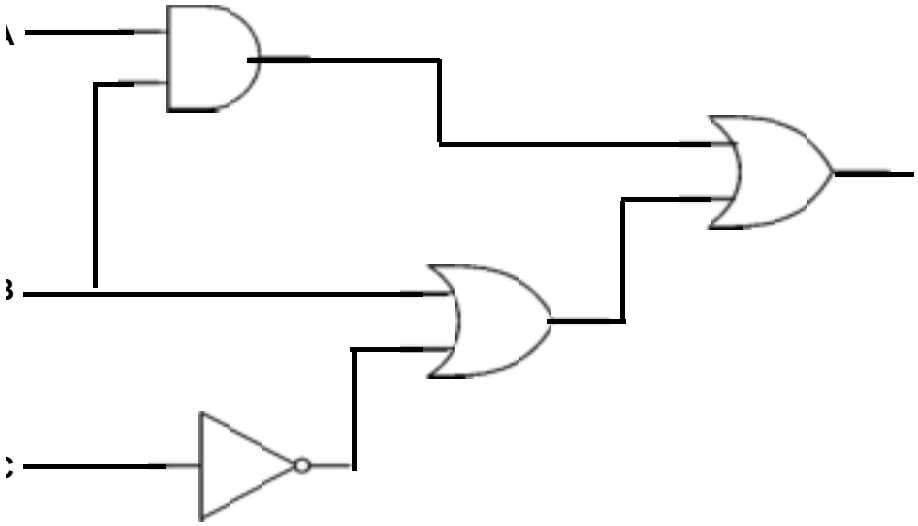
(b) 1 mark for chosen method + 1 mark for description

use of overflow/bucket area used
any record that is subject to a collision is placed, serially, in overflow area
set flag when overflow / bucket is in use

use of linked lists
original location acts as head of list and points to a list of any records that have been subject to a collision

use of next location after occupied one is used
this continues until empty location is found [2]

9 (a) 1 mark for each correct logic gate (accept other logic gate nomenclature)



If a candidate has only one input to AND gate or an OR gate they lose the mark for that gate

[4]

(b)

A	B	C	X
0	*0*	*0*	1
0	*0*	*1*	0
0	*1*	*0*	1
0	*1*	*1*	1
1	*0*	*0*	1
1	*0*	*1*	0
1	*1*	*0*	1
1	*1*	*1*	1

[4]