**CAMBRIDGE INTERNATIONAL EXAMINATIONS**

**GCE Advanced Subsidiary Level and GCE Advanced Level**

# MARK SCHEME for the October/November 2013 series

## 9691 COMPUTING

**9691/11** (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 October/November 2013 series for most IGCSE, GCE Advanced Level and Advanced Subsidiary Level components and some Ordinary Level components.

CAMBRIDGE
International Examinations

**1** 1 mark for method + 1 mark for description
Method must match description

**observation**

– watch system in operation first hand
– find out first hand where there are bottlenecks/problems with existing system

**interviewing**

– interview passengers and staff face to face
– alter questions according to responses given

**questionnaires**

– hand out written questionnaires with pre–tested questions
– avoids need for analyst to be present whilst gathering information

**look at documentation/data flows**

– if any documentation exists, then look through it to see what data needs to be handled by new system
– observe the data flow through the system to find out volumes of data, how data is collected, how data is processed and so on **[4]**

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

– graphics showing the process
– graphics mimic the process accurately/show how process links up
– use colours e.g. green = ON and red = OFF
– use flashing colours/sounds to show faults
– ability to click on graphics to show history/status
– easy to understand/use interface
– ability to easily move between process screens allowing monitoring of all stages in process
– use of sound to warn of problems or to indicate that monitoring is functioning correctly
– use of input devices such as touch screens and trackerballs
– use of output devices such as loud speakers/beepers and printers **[5]**

**(b)** 1 mark for each device and 1 marks for each reason

**touch screen**

– to select (graphics to view status or start/stop parts of process)
– to be able to easily navigate to other screens in the process

**trackerball**

– to select (an item on the screen to change its status)
– to be able to easily navigate to other screens in the process

**keyboard**

– to input values or type in queries/enter commands **[4]**

**3 (a)** **One** point from each:

**verification**

– this is a way of preventing errors when data are copied from one medium to another
– e.g. double entry or visual checks on data
– verification does not check to see if data makes sense or are within acceptable boundaries;
– it only checks to see if original and copied data match exactly

**validation**

– this is a computer check on input data
– see if data matches certain criteria and are reasonable
– example e.g. range check **[2]**

**(b) (i)** length check **[1]**

**(ii)** range check **[1]**

**(c) (i)** **5 4 3 2 1**
**3 0 4 5 ?**

sum = (5x3) + (4x0) + (3x4) + (2x5)
= 15 + 0 + 12 + 10 = **37**
divide 37 by 11 and get 3 remainder **4**

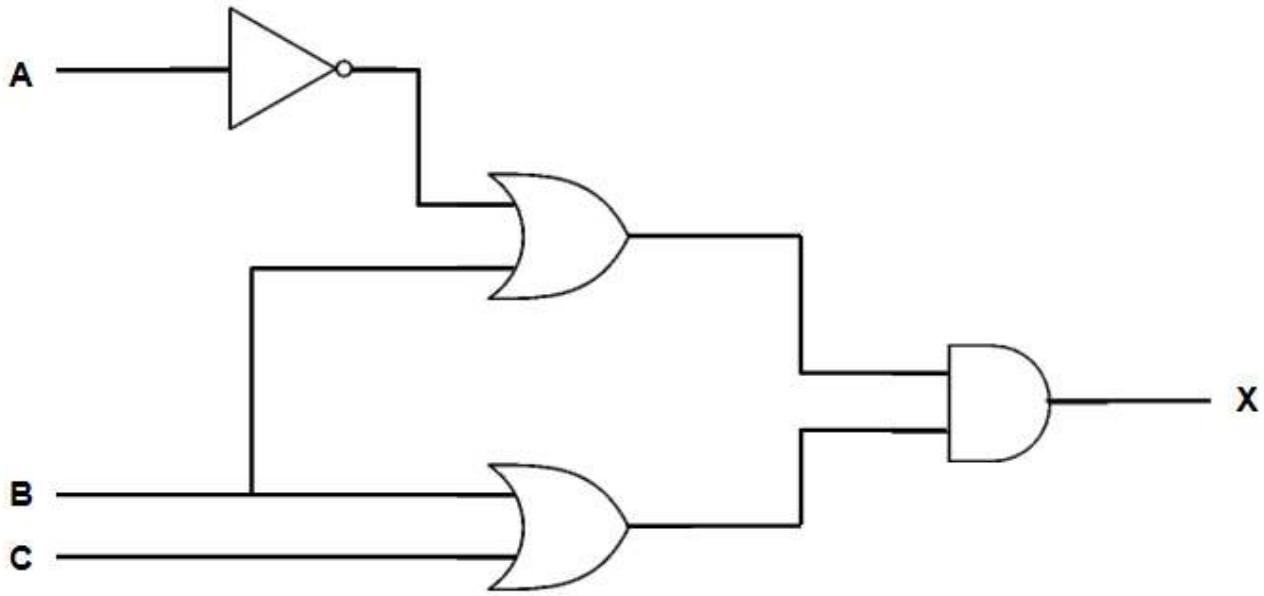
check digit is **4**

(1 mark for a reasonable calculation + 1 mark for correct check digit allow follow through)
**[2]**

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

- check digit for 34921 re–calculated
- check digit value should be 7 (using modulo 11)
- error will be flagged check digit is shown as 1 in transmitted number **[2]**

**4** **(a)** 1 mark for each correct logic gate




**[4]**

**(b)**

| A | B | C | X | |
|---|---|---|---|---|
| *0* | *0* | *0* | **0** | } 1 mark |
| *0* | *0* | *1* | **1** | |
| *0* | *1* | *0* | **1** | } 1 mark |
| *0* | *1* | *1* | **1** | |
| *1* | *0* | *0* | **0** | } 1 mark |
| *1* | *0* | *1* | **0** | |
| *1* | *1* | *0* | **1** | } 1 mark |
| *1* | *1* | *1* | **1** | |

**[4]**

**5 (a) interrupt**

signal sent to the processor (which causes a break in the execution of the current routine) **[1]**

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

- data sent to <u>printer</u> buffer from memory
- buffer is then emptied to printer allowing user to get on editing document
- when buffer empty an interrupt is sent to processor from printer requesting more data
- current job suspended while buffer refills
- use continues editing document whilst buffer emptied contents to printer
- idea of interrupt priorities **[4]**

**(c)**
- response depends on the software package
- produces a dialogue box alerting user that virus has been found inviting user to "clean up" the infected file
- some systems remove viruses automatically and don't request user "permission"
- quarantine the infected file/virus **[2]**

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

- OS will allow one user **at a time** to use the computer
- each approved user is identified by a user id/password
- allows multi-tasking
- provides security for user files/profiles **[2]**

**6 (a) RAM**

- contents can be altered/written to
- holds program/data currently in use
- volatile/temporary memory/contents lost on switching off
- usually has greater memory capacity than ROM

**ROM**

- contents can be read only/can't be altered
- holds bootstrap/BIOS/system data
- non–volatile/permanent memory/retains contents on switching off
- MAX 2 points from ROM and MAX 2 points from RAM **[3]**

**(b) RAM**

- holds the data gathered from the sensors during monitoring
- holds user information entered into system
- holds part of OS/program, in current use

**ROM**

- holds program that is used to control the data logger
- holds the start-up routines when data logger set up for monitoring **[2]**

**(c)** 1 mark for device + 1 mark for application

| Device | application |
|---|---|
| – barcode/QR reader | automatic stock control<br>library systems |
| – rfid | tracking live stock<br>timing athletes automatically in races |
| – biometrics | security systems<br>passports |
| – magnetic stripes | credit/debit cards<br>hotel key cards |
| – OCR | checking signatures on documents |
| – voice recognition | helping blind people interface with a computer system |
| – smart (card) system | loyalty cards<br>passports |
| – OMR | automatically marking exam papers |
| – MICR | banking/cheques **[2]** |

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

- pH and oxygen sensors send data/signal**s** to the data logger
- ADC
- data is stored in data logger's memory
- data is sampled at a set time interval (depending on memory size)
- data logger taken back to laboratory after monitoring period
- data is downloaded to a computer
- data logging software interprets information…
- …and shows graphs of how oxygen and acidity levels changed over the monitoring period
- built in routines warn if levels dangerously high **[4]**

**7 (a)** 1 mark for each benefit and 1 mark for each drawback

| benefit | drawback |
|---|---|
| **bus** | |
| – requires less cabling than the<br>– other topologies<br>– cheapest system to set up | – if there is a fault in the central cable, whole system affected<br>– doesn't work well under heavy loading<br>– less secure |
| **star** | |
| – failure in any connection and network still functions<br>– easier to identify faults<br>– easier to expand network<br>– each node can have different kind of cable<br>– a more secure network | – if central hub fails, whole network fails<br>– more expensive to set up |
| **ring** | |
| – works well under heavy loading<br>– possible to form very large networks | – faulty connection can cause whole network to fail<br>– difficult to expand this type of network<br>– less secure **[6]** |

**(b)** 1 mark per point

**LAN**

– hub
– (cat 5) network cabling
– network interface card (NIC)
– gateway
– server
– bridge
– switch

**WAN**

all the above plus:

– broadband modem
– telephone cabling/radio links/satellite links
– router **[3]**

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

- use of formulas
- inbuilt functions
- automatic graphs produced
- automatic re-calculation of cells
- "what if" scenarios/predictions
- possible to install macros **[3]**

**(b)** 1 mark for each advantage and 1 mark for each disadvantage

**advantages**

- often get better customer support since in direct contact with software designers/writers
- designed and written to meet unique user's specific requirements
- writers of software can develop it so it will not interfere with other software used in the company
- users get exactly what they want – therefore usually more user-friendly
- any modifications needed later on can be done more easily and quickly
- other companies cannot use the software (for the company alone)

**disadvantages**

- much greater dependence on software company (could go out of business)
- unlikely to be as well developed as off-the-shelf products
- usually much more expensive since all development costs have to be met by the customer
- development time may be very lengthy
- can be a gamble since final package may not exactly meet user requirements in a reasonable time scale
- no user groups or eq. **[4]**

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

- specialist control applications (e.g. nuclear plants)
- specialist monitoring applications (e.g. chemical process)
- modelling tasks
- virtual reality applications
- hospital/medical monitoring
- air traffic control **[2]**

**9 (a)** – mimics knowledge and experience of human experts
– uses knowledge base and inference engine to solve problems in a "human like" manner **[1]**

**(b)** 1 mark per point:
**Cannot get max unless use all three parts**

**inputs**

– engineer inputs the fault(s)
– answers on screen questions that are based on previous responses
– uses an interactive interface

**processes**

– expert system analyses data
– inference engine compares data/makes reasoned conclusions…
– …with that held in/based on data in knowledge base
– …uses the rule(s) base/set of inference rules
– until matches are found

**outputs**

– percent probability of fault
– how to fix the problem
– outputs reasons it produces a particular answer **[7]**