UNIVERSITY OF CAMBRIDGE INTERNATIONAL EXAMINATIONS

GCE Ordinary Level

MARK SCHEME for the October/November 2010 question paper for the guidance of teachers

7010 COMPUTER STUDIES

7010/11 Paper 1, maximum raw mark 100

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 must be read in conjunction with the question papers and the report on the examination.

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

CIE is publishing the mark schemes for the October/November 2010 question papers for most IGCSE, GCE Advanced Level and Advanced Subsidiary Level syllabuses and some Ordinary Level syllabuses.

1 (a) check digit

– validation check
– single digit appended to a number
– calculated from digits and their position
– re-calculated after data transfer

– e.g. bar codes, ISBN, credit/debit cards [2]

(b) RAM

– random access memory
– memory lost on switching off/volatile/temporary
– stores user programs/data (etc.)
– usually on a chip
– can be read/changed by user

e.g. SRAM,DRAM etc. [2]

(c) macro

– macro instruction
– new command created by combining number of existing ones
– can combine effects of pressing several individual keys on k/board
– can be programmed by user to customise software
– e.g. single key stroke to insert a logo into a document [2]

(d) USB flash memory

– (memory data) storage device
– removable/portable
– uses universal serial bus connector
– re-writable device
– contains printed circuit board
– allows transfer of data/files between computers
– draws power from the computer port
– contains EEPROM (electrically erasable programmable ROM)/ non-volatile memory
– e.g. pen drive/memory stick/thumb drive

[2]

(e) printer buffer

– temporary storage/memory
– compensates for the difference in speed of printer and CPU
– e.g. holds data whilst computer completes a job, recovering from error (e.g. paper jam)

[2]

2 **(a)** Any **three** from:
- "glitches in the software" e.g. divide by zero
- software conflicts
- virus
- operating system software loss/corruption
- hardware malfunction (e.g. overheating of circuit board, processor fans failing etc.)
- hardware incompatibility
- power supply interruption/"spikes"
- incorrect power down after use
- hard disk crash/failure

[3]

(b) Any **one** from:
- Grandfather-Father-Son (GFS)/file generation system
- backups
- parallel systems
- type/scan and OCR in new data again from the hard copies [1]

(c) Any **one** from:
- encryption
- encrypt files

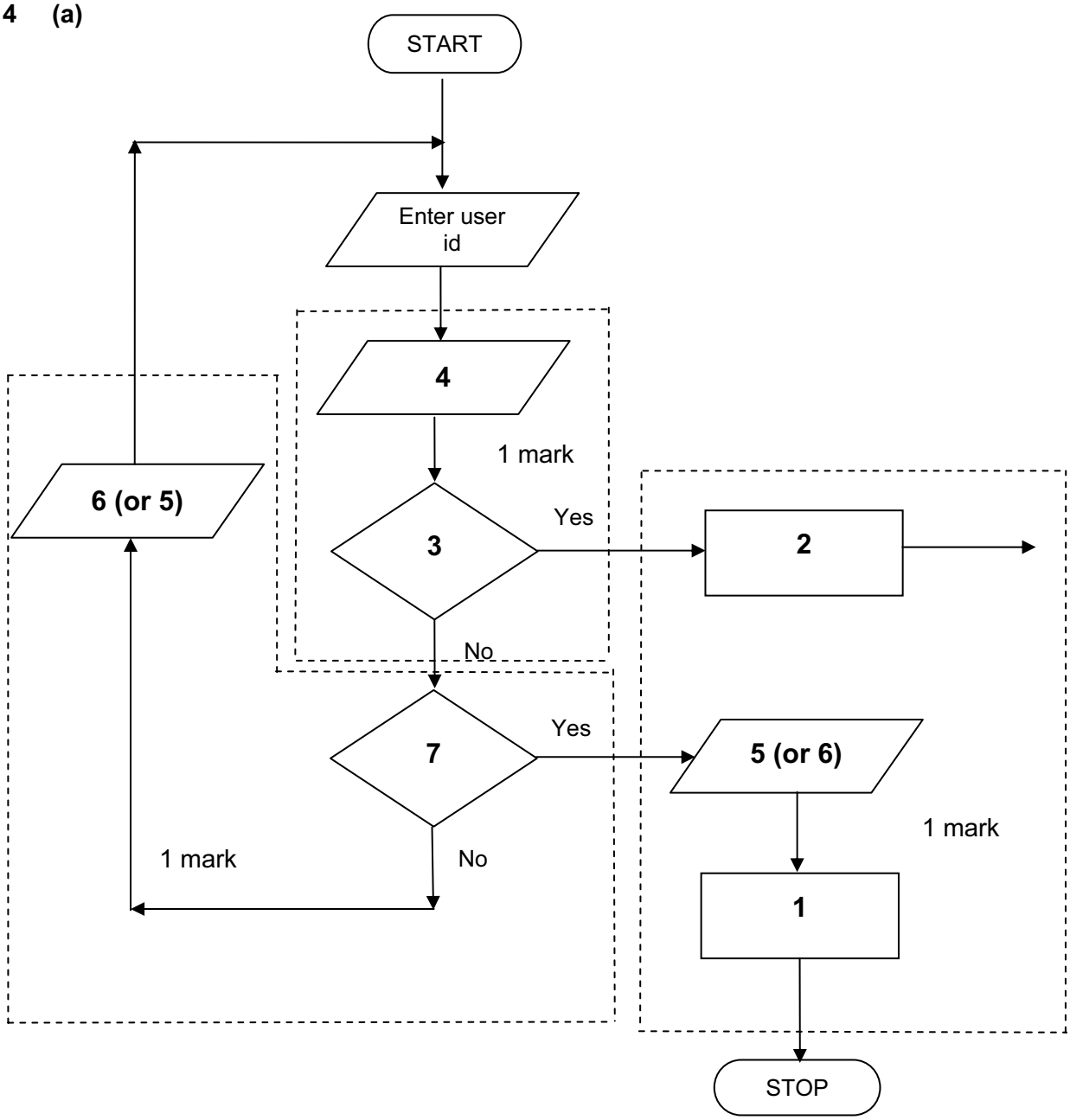
[1]

3 **(a)** STAR, BUS [2]

(b) Any **one** from:
- can use any station to access files, etc.
- can share files etc.
- can share resources (e.g. printer)
- allows easier communication between users [1]

(c) Any **one** from:
- more easily/more rapid transfer of viruses from computer to computer
- file (etc.) security is more difficult
- extra infrastructure costs e.g. cabling [1]

4 (a)



1 Access not allowed
2 Allow access
3 Do user id and password match
4 Enter password
5 Error message
6 Error message
7 Three attempts [3]

(b) verification [1]

5 **(a)** 2 marks (max) for RTTP points; 2 marks (max) for RTPC points

real time transactions	real time processing
– individual transaction processed as it occurs	– physical quantities continuously monitored – inputs compared with pre-set values
– files/fields/records updated immediately	– processed fast enough to affect input
	– uses sensors, ADC, DAC, etc.
– e.g. online booking of seats	– e.g. temperature control in air con [4]

(b) Any **two** points from:
- file management
- input/output control
- spooling
- memory management
- multiprogramming
- multitasking/JCL/batch processing
- handling interrupts
- error reporting/handling
- security (e.g. virus checking)
- user interface (e.g. WIMP)
- processor management
- loads/runs programs
- user accounts
- utilities [2]

6 **(a)** Any **one** from:
- reduced costs (no/less printing, no/less distribution of directories)
- faster/easier updating procedure
- raising profile of company [1]

(b) Any **two** from:
- faster/easier to find information
- more accurate/up-to-date
- more information/data available
- could easily extend to international directories [2]

(c) Any **one** from:
- more likely to get calls from call centres/sales companies
- unsolicited calls
- mis-use of details [1]

(d) Any **one** from:
- number changed and not registered
- errors in the information [1]

7 **(a)** **(i)** Any **one** from:

– interview customers
– hand out questionnaires to customers [1]

(ii) 1 mark for method and 1 mark for reason:

– DIRECT
– must have only one way of conveying/updating the information

– PILOT
– could adopt new system at one terminal only to trial new system

– PARALLEL
– Check new system is working correctly/back up in case of system failure [2]

(b) Any **one** from:
– current time
– terminal number/name
– date
– baggage reclaim/carousel number
– name of airline
– transfers/connections [1]

(c) Any **one** from:
– touch screens/touch pad/mouse/tracker ball [1]

(d) Any **two** from:
– fewer errors
– could be linked to website for live updates
– faster/more accurate updating of information
– no language problems for customers
– no need to wait in a queue at manned help desks [2]

8 **(a)** 1 mark for hardware and 1 mark for software:

<u>hardware</u>
– webcam
– microphone
– large TV/monitor/screen
– router/broadband modem
– communications cables
– speakers

<u>software</u>
– compression software/CODEC
– communications software [2]

(b) Any **two** from:
- language differences
- time differences
- controlling a 3-way conversation
- possible poor communications/loss of connection/slow connection
- delay in transmission [2]

(c) Any **two** from:
- less time lost in travelling
- can hold meetings with little notice
- safer (***must be qualified*** e.g. terrorism risk, less travelling, etc.)
- can involve more people company-wide [2]

9 **1 mark** for each error and **1 mark** for reason why it is an error

- line 1/negative=1 and/or line 2/positive=1

- negative and/or positive should be set to zero

- line 7/count=count+1
- don't need a count within a **for …. to next** loop
- replace loop with a **repeat...until** loop

- line 8/**print** negative, positive or line 9/**next** count
- outputs should come after the **next count** statement [6]

10 **(a)** 6 (fields) [1]

(b) 3002, 2002, 3003, 3004 [2]

(c) (Length (m) > 74) OR (Max Speed (kph) < 900)

← - (1 mark) - → ← - - - - - - (1 mark) - - - - - - →

OR

(Max Speed (kph) < 900) OR (Length (m) > 74)

← - - - - (1 mark) - - - - → ← - - - (1 mark) - - - → [2]

11 **(a)** Any **three** points from:
- (count) number of vehicles …
- … at various times of day/at different positions/in different directions
- put data into computer …
- … and try out different scenarios
- look at effect of accidents/break downs
- look at effect of heavy traffic
- determine optimum timings of lights
- effect of emergency vehicles/public transport [3]

(b) Any **two** from:

- less expensive (***must be qualified)***
- much safer prevents accidents/traffic problems through incorrect lighting times
- can try out many scenarios first (to give optimum settings)
- much faster than doing actual "experiments" on real lights [2]

(c) Any **two** from:

- sensors detect cars at each junction
- sends signals/data to computer
- computer software counts number of cars
- if analogue data, need an ADC
- compares sensor data with stored data/simulation results
- changes light timings/sequences as required
- (uses DAC) to send signals back to lights (control)
- continuously monitors [2]

12 **(a)** = SUM(B2:M2)/12 OR
= AVERAGE(B2:M2) OR
= (B2+C2+D2+E2+F2+G2+H2+I2+J2+K2+L2+M2)/12
[rounded] [1]

(b) = (L5 – L4) * L3 (must use cell references) [1]

(c) **(i)** graph "B" since rainfall usually measured as a height/bars
graph "B" since the information is clearer [1]

(ii)
- draw a line at value 8
- include a row with all values 8 and add this data [1]

(d) Any **two** from e.g.
- weather forecast for 7/14 days
- attractions/facilities in the area
- online booking e.g. hotels
- maps/how to get there
- buttons linking to other web pages/site
- videos/multimedia presentations
- search facility
- images of resort/virtual tours [2]

13 Any **four** from:
- collect information from expert(s)
- put information into the/create knowledge base
- develop YES/NO dialogue/user interface
- output screens designed
- fully tested with known expected outputs
- produce user manuals
- fully train users of the system
- reference to inference engine being created
- reference to rules base being created [4]

14 (a) **delete**

- customer leaves the bank/close account
- customer dies

amend

- change of address
- change of telephone number
- change account details
- change name after marriage
- transactions on account e.g. deposits, withdrawals

insert

- new customer joins bank/opens new account [3]

(b) (i) Any **one** from:
- saves memory/less space required on the file
- faster/easier to type in
- faster to search for information
- fewer errors [1]

(ii) 1 mark for name, 1 mark for reason and 1 mark for improvement

- AGE
- always changing
- need to keep updating each year
- date of birth [3]

15 EACH RESPONSE MUST BE DIFFERENT

(a) (i) Any **one** from:
- character/type check
- length check
- Boolean check
- presence check

(ii) Any **one** from:
- format check
- character/type check
- length check
- presence check

(iii) Any **one** from:
- range check
- character/type check
- presence check [3]

(b) Any **one** from:
- drop down lists showing M or F only, possible dates, etc.
- use of touch screens with only certain data options
- use of restricted lists
- radio buttons [1]

(c) **(i)** Any **one** from:
- lock computer
- log off the system
- if in an office, lock the door
- put into sleep/hibernate mode with password [1]

(ii) Any **one** from:
- to prevent RSI
- to prevent neck/back problems possible
- to prevent eye sight problems/headaches [1]

16 **(a)** Any **three** from:
- satellites transmit signals to computer/sat nav in car
- sat nav system in car receives these signals
- depends on **very** accurate time references/atomic clocks
- **each** satellite transmits data indicating location and time
- sat nav system car calculates position based on at least 3 satellites
- at least 24 satellites in operation world wide
- sat nav system combines satellite information with mapping info [3]

(b) Any **two** from:
- no need to read/own maps
- driver doesn't need to memorise route
- can give useful information such as location of garages/speed cameras/points of interest/traffic congestion
- allows driver to concentrate on driving (therefore safer)
- can find shortest/fastest route
- easier to re-route in case of road closures, etc.
- updateable [2]

(c) Any **one** from:
- stored maps out of date (instructions go to incorrect roads)
- inaccurate positioning
- loss of signal
- errors in original data/setting up
- sends vehicles down inappropriate routes
- over reliance by driver on the sat nav [1]

(d) Any **one** from:
- ships
- aeroplanes [1]

17 **Marking Points**

- initialisation of running totals (1 mark)
- correct loop control (1 mark)
- error trap for height input (1 mark)
- error trap for weight input (1 mark)
- sum total1 and average1 (i.e. height) calculation (1 mark)
- sum total2 and average2 (i.e. weight) calculation (1 mark)
- correct output (only if some processing attempted, must be outside loop) (1 mark)

[max: 5]

Sample pseudocode

```
total1 = 0: total2 = 0                                          (1 mark)

for x = 1 to 1000                                               (1 mark)

    input height, weight

        if height > 2 or height < 0 then print "error": input height   (1 mark)

        if weight > 130 or weight < 0 then print "error": input weight  (1 mark)

            else total1 = total1 + height: total2 = total2 + weight

next x

average1 = total1/1000                                          (1 mark)

average2 = total2/1000                                          (1 mark)

print average1, average2                                        (1 mark)   [5]
```