CAMBRIDGE INTERNATIONAL EXAMINATIONS

Cambridge International Advanced Level

MARK SCHEME for the May/June 2015 series

9608 COMPUTER SCIENCE

9608/31 Paper 3 (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 2015 series for most Cambridge IGCSE®, Cambridge International A and AS Level components and some Cambridge O Level components.

CAMBRIDGE
International Examinations

1 (a) (i)	';' missing	1
(ii)	'2' is not a variable	1
(iii)	'e' is not a valid letter	1
(b)	<assignment statement> ::= <variable> = <variable><operator><variable>; <variable> ::= <letter>|<letter><letter> |<letter><letter><letter> <letter> ::= a|b|c|d <operator> :: =+|-|*|÷	2 2 1 1
(c)	<letter> | <letter><variable> // <letter> | <variable><letter>	2

(d) (i)	debugging is faster / easier
	// can debug incomplete code
	// better diagnostics
		1

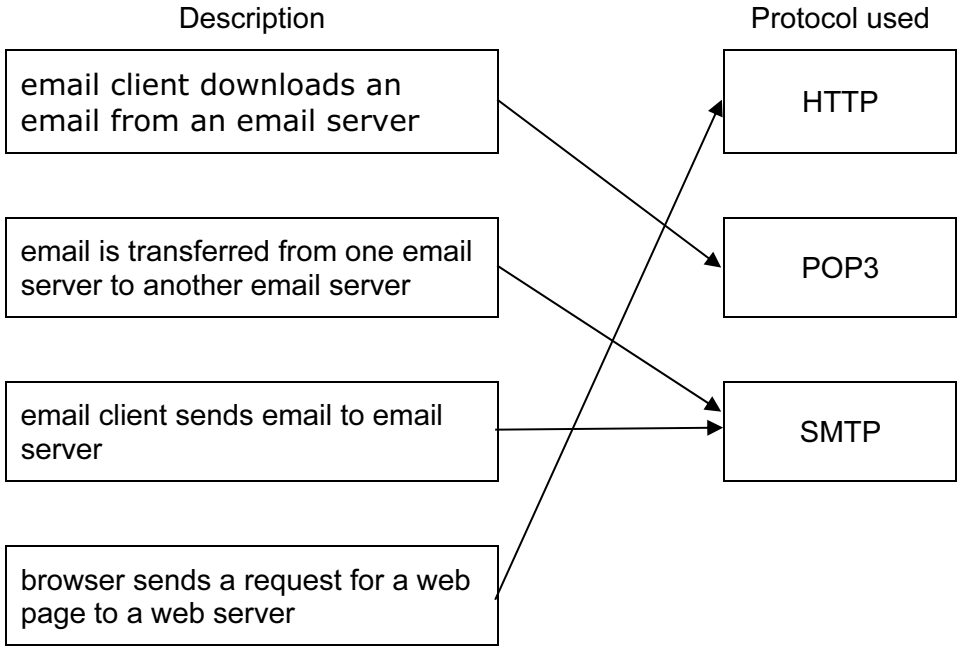
(ii)	compiler produces executable version – not readable / no need for source code // difficult to reverse-engineer	1
		Total: 13
2 (a)	• Spam • Worm **Pharming** redirect website to fake website // domain name server compromised // proxy server compromised **Phishing** through email attempt to obtain somebody's confidential data / install malware	1 1 1 1
(b)	**Spam** • user's inbox is filled by large amount of unwanted email • user / email server employs filtering software that can divert / delete spam email **Worm** • could corrupt user's computer // delete data // consume bandwidth • run anti-virus software in the background // not connect to the Internet // keep OS up-to-date	1 1 **or** 1 1

(c)	**encryption**: process of turning plain text into cipher text **public key**: key widely available that can be used to encrypt message that only owner of private key can decrypt // can be used to decrypt a message thereby confirming originator of message	1 1
(d) (i)	digital signature	1
(ii)	• software is put through hashing algorithm • hash total is encrypted with private key (digital signature) • software + encrypted hash / digital signature are sent • receiver is in possession of sender's public key • the received hash total / digital signature is decrypted with public key (SH) • the receiver hashes received software (RH) • If SH matches RH then software is authentic and has not been altered	Any **four** points 1 mark each
		Total: 13
3 (a) (i)	enumerated	1
(ii)	record	1
(iii)	`MyMonthOfBirth ← DateOfBirth.ThisMonth`	1

(b) (i)	`TYPE LocationRainfall`
	`  DECLARE LocationName : STRING`
	`  DECLARE LocationHeight : INTEGER`
	`  DECLARE TotalMonthlyRainfall : ARRAY[1..12] OF REAL`
	`ENDTYPE`
		1
		1
		1
		1 +1

(ii)	• no need to re-sort data every time new data is added • only a small file so searching will require little processing • new records can easily be appended	1 1 1 [max 2]
		Total: 10
4 (a) (i)	<table><tr><th colspan="3">Circuit 1</th></tr><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	1

(ii)	<table><tr><th colspan="3">Circuit 2</th></tr><tr><th>A</th><th>B</th><th>X</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	1
(b) (i)	• circuit 1: $\overline{A.B}$ • circuit 2: $\overline{A} + \overline{B}$	1 1
(ii)	$\overline{A.B} \equiv \overline{A} + \overline{B}$	1
(c)	$\overline{\overline{(A+B)}.B}$ Mark as follows: $\overline{(A+B)}$ $.B$ bar over whole expression	 1 1 1
(d)	$\overline{\overline{(A+B)}.B}$ $= \overline{\overline{(A+B)}} + \overline{B}$ $= (A+B) + \overline{B}$ $= A + (B + \overline{B})$ $= A + 1$ $= 1$ allow f.t. from **(c)**	 1 1 1 1 1 [max 3]
		Total: 11
5 (a)	Monitoring system	1
(b)	• temperature sensor … • transmits measured temperature • analogue to digital converter … • converts analogue signal from sensor to digital value that can be stored • storage device // data logger… • for recording readings from sensor • transmission hardware … • to transfer data from sensor to storage device • processor … • to process incoming data	1 1 1 1 1 1 1 1 1 1 [max 6]

(c) (i)	temperature reading in location 5 has been processed	1 1
(ii)	0100 0000 1111 1011 1 mark per byte	2
(d) (i)	`AND #B00010000 // AND #&10 // AND #16` 1 mark for `AND`, 1 mark for address mode, 1 mark for mask, 1 mark for indication of numbering system	1 + 1 + 1 + 1
(ii)	`OR #B00000001 // OR #&01 // OR #1` 1 mark for `OR`, 1 mark for mask	1 +1
		Total: 17
6 (a)		1 mark for correct arrow from each description
(b)	peer-to-peer	1

(c) (i)	**Tracke**r: central server that:
	stores details of other computers that have all / part of file to be downloaded
	// has data on those peers downloading and uploading file
	// shares IP addresses with other clients in swarm allowing them to connect
		1
		
		1

(ii)	**Seed**: peer computer that has 100% of file // is uploading downloaded content	1 1
(iii)	**Swarm**: all the connected peer computers that have all or part of the file to be downloaded / uploaded // share a torrent	1 1
		Total: 11