#### **CAMBRIDGE INTERNATIONAL EXAMINATIONS**

**Cambridge International Advanced Level** 

# MARK SCHEME for the October/November 2015 series

# 9608 COMPUTER SCIENCE

9608/32

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 October/November 2015 series for most Cambridge IGCSE<sup>®</sup>, Cambridge International A and AS Level components and some Cambridge O Level components.



Page 2	Mark Scheme	Syllabus	Paper
	Cambridge International A Level – October/November 2015	9608	32

# **1** (a) (i) 01101000 0011

 $= 0.1101 \text{ (or } 1/2 + 1/4 + 1/16) \times 2^{1}3$  = 110.1 = 6.5[1+1]

(ii) +3.5 = 11.1 [1] = 0.111 × 2†2 (or indication of moving binary point correctly) [1] = 01110000 0010

 (iii)
 01110000
 Allow f.t. from (ii)

 10001111
 One's complement on mantissa
 [1]

 10001111 +1
 Two's complement
 [1]

= 10010000 0010 [1]

(b) (i) Precision/accuracy of numbers represented will increase [1]

(ii) Range of numbers represented will increase [1]

(c) Any point, 1 mark (max. 3)

0.1/0.2 cannot be represented exactly in binary // rounding error

0.1 represented by a value just greater than 0.1 // 0.2 represented by a value just greater than 0.2

adding two representations together adds the two differences

summed difference significant enough to be seen

[1]

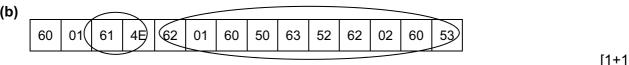
[max. 3]

[Total: 14]

2 (a)

Symbol	Token				
Зушьы	Value	Туре			
Start	60	Variable			
0.1	61	Constant			
Counter	62	Variable			
10	63	Constant			

Page 3	Mark Scheme	Syllabus	Paper
	Cambridge International A Level – October/November 2015	9608	32



(b) _				_												
	60 0	1 61	45	62	01	60	50	63	52	62	02	60	53			
															[	1+1]
(c) (i	i) syr	itax ana	alysis	<b>;</b>												[1]
(i	i) any	/ <b>two</b> p	oints	from	:											
	che	nstruct   ecking s duce e	synta	x/gra	mma		1								[ma	x. 2]
(d) (i	i) Mir	nimise t	:he <u>e</u>	<u>kecut</u>	<u>ion</u> ti	me //	cod	<u>e</u> run	s fas	ter						[1]
(ii	i) Co	mpiler o	could	calcı	ulate	2*6	and ı	repla	ce it	with 1	he v	alue	12.			[1]
(iii	AD STO AD	D 436 D 437 O 612 D 438 O 613													} }	[1] [1] [1]
	<b>–</b> 1	for eac	h add	dition	al ins	struct	ion; (	0 for	copy	of o	rigina	al cod	de		[Total	: 13]
		ed circu asts for						1								[1] [1]
р с с с с с	s: give s: split s: sen s: who s: fast s: pacl s: pacl s: bett s: pac	es dedic into pa ds pack le band er data kets arr kets ca er for a	ackets kets of dwidtl trans rive ir nnot real- ay arr	s/chu on inc h ava sfer n orde get lo time rive o	nks lividu ilable er the est appli ut of	e // p ey are icatio orde	s: sh e ser n er so	nt delay	/ unti	l pac	ket c		resto	ored		[1] [1] [1] [1] [1] [1] [1]

3

(c) web page divided into packets/chunks	[1]
each packet has destination address	[1]
router looks at IP address	[1]
and decides where to send packet next for most efficient path	[1]
packets can take different routes	[1]
home computer reassembles packets to rebuild web page	[1]
	[max. 3]

[1]

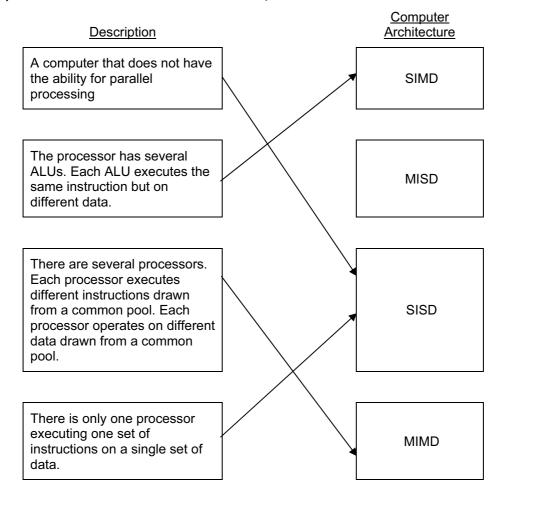
[max. 6]

ps: packets may get lost so retransmission causes delays

Page 4	Mark Scheme	Syllabus	Paper
	Cambridge International A Level – October/November 2015	9608	32

[Total: 11]

4 (a) 1 mark for correct arrow from each description



- (b) (i) Massive: many/large number of processors // hundreds/thousands of processors [1]
  - (ii) Parallel: to perform a set of coordinated computations in parallel/simultaneously [1]
- (c) processors need to be able to communicate ... [1] so that processed data can be transferred from one processor to another [1]
  - suitable algorithm/program/software/design // appropriate programming language [1] which allows data to be processed by multiple processors simultaneously [1]

[Total: 10]

[4]

Page 5	Mark Scheme	Syllabus	Paper
	Cambridge International A Level – October/November 2015	9608	32

5 (a) (i)

$$Z=P.\overline{Q}.\overline{R}$$
 +

 $P.\overline{Q}.R + [1]$ 

P.Q.R [1]

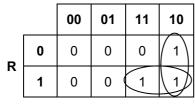
(ii)

		PQ					
		00	01	11	10		
R	0	0	0	0	1		
ĸ	1	0	0	1	1		

[1]

(iii) 1 mark each loop

PQ



Allow f.t. from (ii) [2]

(iv)

$$Z=$$
 $P.\overline{Q}$ 
[1]

+P.R [1]

Allow f.t. from (iii)

(b) (i) 1 mark row headings. 1 mark column headings. 1 mark per 2 correct rows (based on headings)

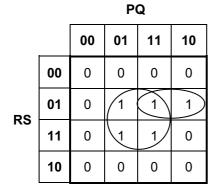
PQ

		00	01	11	10
	00	0	0	0	0
RS	01	0	1	1	1
KO	11	0	1	1	0
	10	0	0	0	0

[4]

Page 6	Mark Scheme	Syllabus	Paper
	Cambridge International A Level – October/November 2015	9608	32

(ii) 1 mark for loop with two 1s; 1 mark for loop with four 1s



Allow f.t. from (i

-1 for each incorrect grouping, max. 2 errors [2]

(iii)

$$Z = Q.S + P.R.\overline{S}$$
 [1]

Allow f.t. from (ii). -1 error if more than 2 terms

[Total: 16]

## 6 (a) blocked → ready:

process is waiting for resource/I/O operation to complete (blocked state)

when I/O operation completed process goes into ready queue (ready state)

running → ready:

when process is executing it is allocated a time slice (running state) // process is allocated time on processor

when time slice completed/interrupt occurs process can no longer use processor even though it is capable of further processing (ready state)

[1]

- (b) to be in blocked state process must initiate some I/O operation to initiate operation process must be executing [1] if process in ready state cannot be executing/must be in running state [1]
- (c) (i) exit/termination/completion [1]
  - (ii) when the process has finished execution [1]

### (d) low-level scheduler:

decides which of the processes in ready state
should get use of processor/be put in running state
based on position/priority
invoked after interrupt/OS call

[1]
[max. 2]

[Total: 11]