OCR
RECOGNISING ACHIEVEMENT

GCE

Computing

Advanced GCE

Unit **F453:** Advanced Computing Theory

Mark Scheme for January 2011

Oxford Cambridge and RSA Examinations

OCR (Oxford Cambridge and RSA) is a leading UK awarding body, providing a wide range of qualifications to meet the needs of pupils of all ages and abilities. OCR qualifications include AS/A Levels, Diplomas, GCSEs, OCR Nationals, Functional Skills, Key Skills, Entry Level qualifications, NVQs and vocational qualifications in areas such as IT, business, languages, teaching/training, administration and secretarial skills.

It is also responsible for developing new specifications to meet national requirements and the needs of students and teachers. OCR is a not-for-profit organisation; any surplus made is invested back into the establishment to help towards the development of qualifications and support which keep pace with the changing needs of today's society.

This mark scheme is published as an aid to teachers and students, to indicate the requirements of the examination. It shows the basis on which marks were awarded by Examiners. It does not indicate the details of the discussions which took place at an Examiners' meeting before marking commenced.

All Examiners are instructed that alternative correct answers and unexpected approaches in candidates' scripts must be given marks that fairly reflect the relevant knowledge and skills demonstrated.

Mark schemes should be read in conjunction with the published question papers and the Report on the Examination.

OCR will not enter into any discussion or correspondence in connection with this mark scheme.

Any enquiries about publications should be addressed to:

OCR Publications
PO Box 5050
Annesley
NOTTINGHAM
NG15 0DL

Telephone: 0870 770 6622
Facsimile: 01223 552610
E-mail: publications@ocr.org.uk

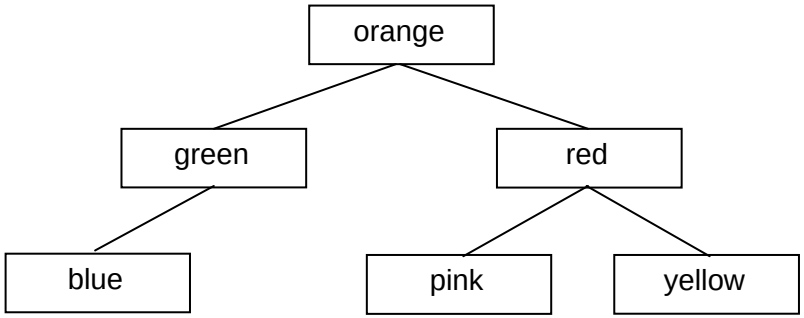
One mark per bullet point unless indicated otherwise

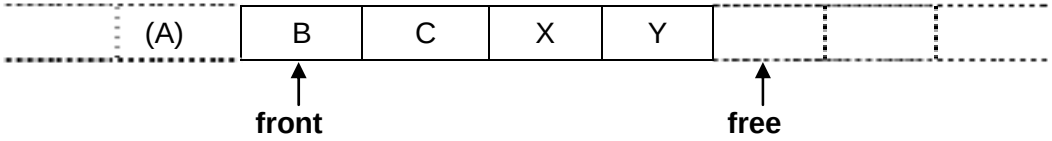
Question	Expected Answer	Mark	Additional Guidance
1 (a)	• output data to disk drive/storage device • for printing at another time • to allow sharing/on a network • job **references** stored in a queue/buffer • avoids delays / avoids speed mismatch • as printers are relatively slow • jobs can be prioritised	**max [4]**	
(b) (i)	• ways of partitioning memory • allow programs to run despite insufficient memory/used for virtual memory • segments and pages are stored on backing store • segments and pages are assigned to memory when needed	**max [2]**	Allow mark for: both use indexes
(ii)	• segments are different sizes but pages are fixed size • segments are complete sections of programs, but pages are made to fit sections of memory • segments are logical divisions, pages are physical divisions	**max [1]**	
(iii)	• disk threshing • more time spent swapping pages than processing • computer may 'hang'	**max [2]**	

Question	Expected Answer	Mark	Additional Guidance
2 (a)	• lexical analysis • code generation	**max [2]**	cao
(b)	• accepts output from lexical analysis • **statements/arithmetic expressions/tokens** are checked... • …against the rules of the language/valid example given eg matching brackets • errors are reported as a list (at the end of compilation) • diagnostics may be given • (if no errors) code is passed to code generation • further detail is added to the symbol table… • …eg data type /scope/address	**max [5]**	
(c)	• simplified code / partly translated code… • …which can be run on any computer/virtual machine/improves portability… • …using an interpreter • sections of program can be written in different languages • runs more slowly than executable code	**max [3]**	Accept: (Syntax) error free

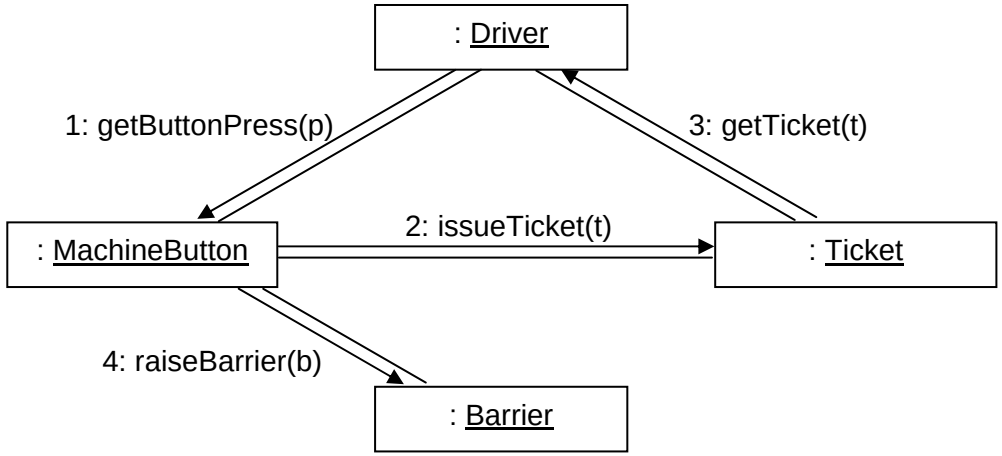
Question	Expected Answer	Mark	Additional Guidance
3 (a) (i)	• a location in the processor • used for a particular purpose • (temporarily) stores data/or control information • explained example of contents held by named register	**max [2]**	
(ii)	• program counter • memory address register • memory data register/memory buffer register • current instruction register • index register • interrupt register • accumulator	**max [2]**	*[Accept status register, although not in the specification]* No abbreviations
(b)	*advantages:* • allows faster processing • more than one instruction (of a program) is processed at the same time • different processors can handle different tasks/parts of same job *disadvantages:* • operating system is more complex... • …to ensure synchronisation • program has to be written in a suitable format • Program is more difficult to test/write/debug	**max [5]**	[max 3 for advantage or disadvantage, total max 5]

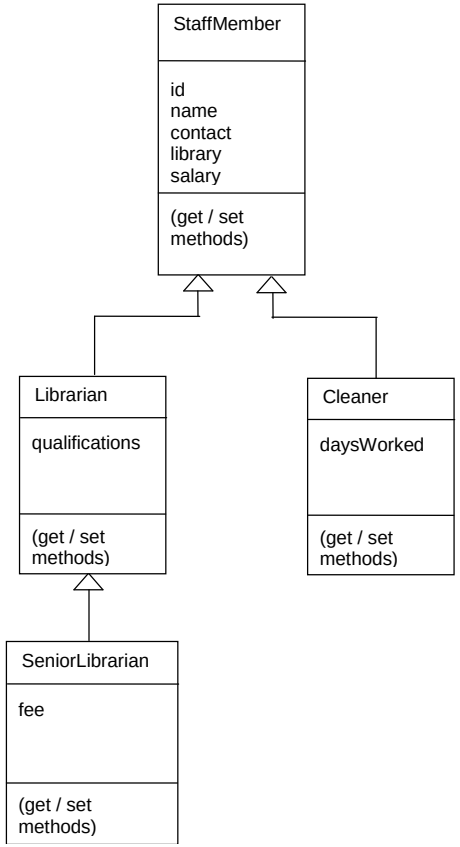
Question	Expected Answer	Mark	Additional Guidance
4 (a)	• P: mantissa 0.010, exponent 010 = 2, P has binary value 1, denary value 1 • Q: mantissa 0.001, exponent 011 = 3, Q has binary value 1, denary value 1 • R: mantissa 0.0001, exponent 100 = -4, R has binary value 0.00000001, denary value 1/256 • R represents a different value *[dependent on P=Q]*	**max [4]**	
(b) (i)	• (exponent is adjusted so that mantissa) starts 01 or 10	**max [1]**	
(ii)	*answer is 01000 001* • (mantissa 00010, move point 2 places) 01000 • (exponent) 001	**max [2]**	
(c)	• maximum (positive) number in this format • as mantissa & exponent each have their largest values	**max [2]**	
(d)	*[answer is 01001 010]* • 2.25 = 010.01 in pure binary • move point 2 places to left… • mantissa is 01001 • exponent is 010	**max [3]**	[*accept any valid method, max 1 if no working shown*, max 3]

Question	Expected Answer	Mark	Additional Guidance
5 (a) (i)	• **size** changes as data is added & removed/**size** is not fixed	**max [1]**	
(ii)	• more complex program to write	**max [1]**	*Accept "more difficult"*
(iii)	• array/fixed length record	**max [1]**	
(b)		**max [3]**	*Accept mirror image* [*award 1 mark each for root, left subtree, right subtree*, max 3]

Question	Expected Answer	Mark	Additional Guidance
(c)	*eg* start at root repeat compare new data with current data if new data < current data, follow left pointer else follow right pointer until pointer is null write new data create (null) pointers for new data *marks for* • start at root • repeat until loop/while loop • comparison of values • follow pointers • condition for end of loop • writing data • creating new pointers • assume new data item is not already in binary tree / is same data type	**max [6]**	Steps should be to a generic binary tree, not a specific example. Allow follow through from (b) if left and right are swapped
(d) (i)	• queue	**max [1]**	
(ii)	 *marks for* • front pointer moved to B • X in cell to right of C **and** Y in cell to right of X • free pointer moved to correct position	**max [3]**	

Question	Expected Answer	Mark	Additional Guidance
6 (a)	• imperative language • uses sequence, selection & iteration • program states what to do… • … & how to do it • program statements are in blocks • each block is a procedure or function • logic of program is given as a series of procedure calls	**max [4]**	
(b) (i)	• getButtonPress( )/processKeyMessage( )/unlock( )	**max [1]**	*Allow answer with no brackets or with (n)*
(ii)	• Driver/Key/Car	**max [1]**	*Allow lower case/preceded by :/underlining*
(iii)	• FlashLights	**max [1]**	*Allow lower case*
(iv)	• object diagram	**max [1]**	
(c) (i)	• communication diagram	**max [1]**	*Accept "collaboration diagram"*

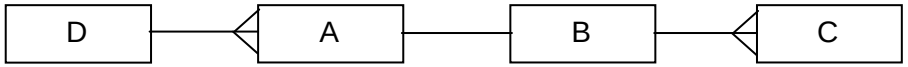
Question	Expected Answer	Mark	Additional Guidance
(ii)	 *marks for* • 2 or more arrows correct • issueTicket( ) label OR getTicket( ) label OR raiseBarrier( ) label • correct number sequence: 2,3,4	**max [3]**	*Give mark for arrows in correct directions even if labels are incorrect* *Give mark for label if brackets are missing*

Question	Expected Answer	Mark	Additional Guidance
(d)	*eg*  *marks for* • superclass StaffMember/Staff/Employee *[accept other relevant name, but not Librarian]* • Librarian as subclass of the named superclass • Cleaner as subclass of the named superclass • SeniorLibrarian as subclass of Librarian • at least 3 of name, contact, library, salary in the superclass only • qualifications in Librarian only • fee in SeniorLibrarian only • daysWorked in Cleaner only • inheritance symbols used correctly throughout • indication of any relevant get or set methods in correct position • all 4 classes included, no extra classes added	**max [8]**	***Must be a class diagram***

Question	Expected Answer	Mark	Additional Guidance
7 (a)	**Mark band 6-8. High level response.** Candidate has explained all 3 of the terms and made some comparisons between them. Candidate has used appropriate technical terminology throughout. There are few, if any, spelling errors or grammatical errors. **Mark band 3-5. Medium level response.** Candidate has explained all 3 of the terms, or explained 2 of the terms and made some comparisons between them. Candidate has used some technical terminology in the response. There may be spelling errors or grammatical errors, but they are not obtrusive. **Mark band 0-2. Low level response.** Candidate has listed some relevant points but failed to explain the terms or make comparisons. There is a lack of cohesion in the response. Candidate has failed to use correct technical terms in the response. Spelling and grammatical errors affect the readability of the response. *Points may include:* *function:* • block of code… • …which performs a single task/calculation… • returns a single value • uses local variables *procedure:* • block of code… • …which performs a task • …which may or may not produce a single value • uses local variables *stepwise refinement:*		*A distinction must be made between function and procedure to qualify as separate terms*.

Question	Expected Answer	Mark	Additional Guidance
	• breaks a problem into sections… • …which become progressively smaller… • …until each module can be written as a single procedure/function • each module can be tested separately • library routines can be used	**max [8]**	
(b) (i)	• stack	**max [1]**	*cao*
(ii)	• (information about) an item of data… • …supplied to a procedure or function • may be passed by reference or by value • used as a local variable	**max [3]**	

<table>
<tr><th>Question</th><th>Expected Answer</th><th>Mark</th><th>Additional Guidance</th></tr>
<tr><td>8 (a)</td><td>
<table>
<tr><th></th><th>Machine code</th><th>Assembly language</th><th>High level language</th></tr>
<tr><td>Uses mnemonics</td><td></td><td>✓</td><td></td></tr>
<tr><td>Uses only binary (or hexadecimal) code</td><td>✓</td><td></td><td></td></tr>
<tr><td>May use relative addresses</td><td></td><td>✓</td><td></td></tr>
<tr><td>May use local variables</td><td></td><td></td><td>✓</td></tr>
<tr><td>Needs translation before the program can be executed</td><td></td><td>✓</td><td>✓</td></tr>
<tr><td>May be translated into intermediate code</td><td></td><td></td><td>✓</td></tr>
</table>
</td><td>max [6]</td><td>Accept an answer that says that local variables can be used in both assembly and high level languages and that relative addressing is used in machine code.
[1 mark per correct row, max 6]</td></tr>
<tr><td>(b)</td><td>• used in assembly language
• uses data in address field…
• … as a constant
[accept diagram or example showing the above]</td><td>max [2]</td><td></td></tr>
<tr><td>(c)</td><td>• used in indexed addressing
• stores a number used to modify an address…
• … which is given in an instruction
• allows efficient access to a range of memory locations/by incrementing the value in the IR
• eg used to access an array</td><td>max [3]</td><td></td></tr>
</table>

Question	Expected Answer	Mark	Additional Guidance
9 (a)	eg  *marks for:* • A and B linked correctly • B and C linked correctly • A and D linked correctly • no additional entities or relationships and all on one diagram with boxes	**max [4]**	
(b) (i)	• unique identifier (in a table)	**max [1]**	
(ii)	• foreign key links tables (to represent many to one relationship)… • …so that only one record is accessed/to avoid duplicate data • eg primary key from B used as foreign key in C from (a) • primary key is in a table that may contain data not required in another table • eg primary key from C is not used in B and hence cannot be a foreign key	**max [4]**	*[Accept any valid example]*
(c) (i)	• StockNo/Quantity/Price	**max [1]**	*cao*
(ii)	• Stock	**max [1]**	*cao*
(iii)	• lists (values of) attributes StockNo, Quantity & Price in the Stock table • …for all Stock with (quantity) less than 100 remaining… • …in order of Price from highest to lowest/descending order of Price	**max [3]**	
(d) (i)	• CHAR is fixed length • VARCHAR is variable length	**max [2]**	

Question	Expected Answer	Mark	Additional Guidance
(ii)	• line 5 defines DepartmentId as an attribute (of Employee)/DepartmentId is set at 5 characters • line 7 defines DepartmentId as the same attribute in the Department table… • …where it is the primary key • …to link the tables	max [3]	
(e)	• so users can access the data they need • users do not need specialist knowledge • to protect data • to prevent unauthorised access.	max [2]	

OCR (Oxford Cambridge and RSA Examinations)
1 Hills Road
Cambridge
CB1 2EU

OCR Customer Contact Centre

14 – 19 Qualifications (General)
Telephone: 01223 553998
Facsimile: 01223 552627
Email: general.qualifications@ocr.org.uk

www.ocr.org.uk

For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored

Oxford Cambridge and RSA Examinations
is a Company Limited by Guarantee
Registered in England
Registered Office; 1 Hills Road, Cambridge, CB1 2EU
Registered Company Number: 3484466
OCR is an exempt Charity

OCR (Oxford Cambridge and RSA Examinations)
Head office
Telephone: 01223 552552
Facsimile: 01223 552553