

Computing

Advanced GCE F452

Programming Techniques and Logical Methods

Mark Scheme for June 2010

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.

© OCR 2010

Any enquiries about publications should be addressed to:

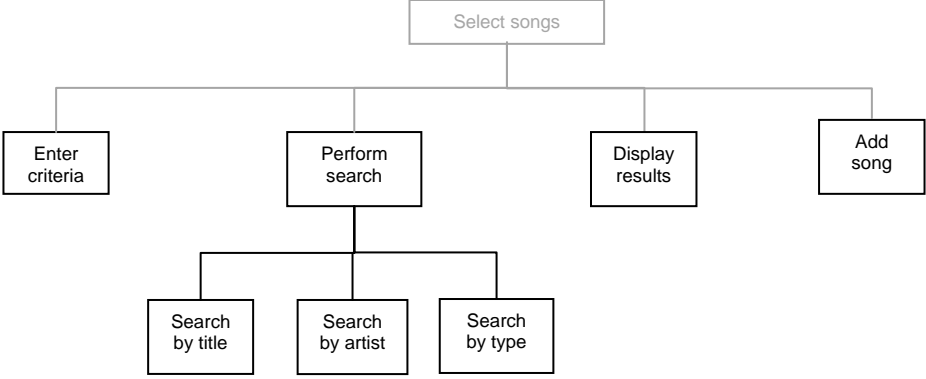
OCR Publications
PO Box 5050
Annesley
NOTTINGHAM
NG15 0DL

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

Question		Expected Answers	Marks	Rationale	
1	(a)	<p>Any 4 of:</p> <ul style="list-style-type: none"> • Used for validation... • ... as it provides an existence check/ to ensure that only teams in the competition can be entered • Prevents typing errors... • ... being entered as a different team • Allows faster input... • ... as operator does not need to type in the name of the team 	[4]		
	(b)	<p>Example:</p> <pre>IF goals_of_first_team > goals_of_second_team THEN points_of(first_team) = points_of(first_team) + 3 ELSEIF goals_of_second_team > goals_of_first_team THEN points_of(second_team) = points_of(second_team) + 3 END IF</pre> <p>Mark points:</p> <ul style="list-style-type: none"> • Correctly, compares the goals scored by both teams (> or <) • 3 points added to first team if it is the winner • 3 points added to second team if it is the winner • Correct nesting/ use of ELSE IF statements <p>[1 per point, max 4]</p>	[4]		
	(c)	(i)	<p>Any 2 of:</p> <ul style="list-style-type: none"> • Different possible inputs are tested... • ... to see if they produce the expected output • How the program works is not considered 	[2]	

Question	Expected Answers	Marks	Rationale
	<p>(ii)</p> <p>Test cases include:</p> <ul style="list-style-type: none"> - second team has more goals than first team - both teams have the same number of goals - the input for number of goals of one of the teams is missing - the input for number of goals of one of the teams is negative - the input for number of goals of one of the teams contains a decimal fraction - the input for number of goals of one of the teams is non-numeric - the input for number of goals of one of the teams is a large integer - the same team has been entered twice - the input for number of goals is extreme/unusually high (test data should be ≥ 5 for this test case to get a mark) <p>Marks: For each test:</p> <ul style="list-style-type: none"> • 1 mark for reason for test (as per test cases above) • 1 mark for test data (provided it matches the reason for test) • 1 mark for expected outcome if it matches test data <p style="text-align: right;">[3 marks per test case x 4: Total 12]</p>	[12]	
(d)	<p>Any 6 of:</p> <ul style="list-style-type: none"> • There is a title • There is a date • All teams are listed in rank/alphabetical order • The rank of each team is shown • The name of each team is shown • The number of points of each team is shown • Suitable layout (columns/column titles/use of space etc) 	[6]	

Question			Expected Answers	Marks	Rationale
2	(a)	(i)	<p>Any 3 of:</p> <ul style="list-style-type: none"> • Each module/task is defined in simple terms... • ... and then split into a number of smaller sub-modules/sub-tasks... • ... which are successively split... • ... until each is small enough to be programmed 	[3]	
		(ii)	<ul style="list-style-type: none"> • Each module is a small part of the problem/focuses on a small sub-task • ... and so easy to solve • ... and test/debug • ... easy to maintain/update_a part of the system ... • ... as the program will be well structured • ... with clearly defined interfaces • ... without affecting the rest • Development can be shared between a team of programmers... • ... so program developed faster... • ... easier to monitor progress... • ... modules can be allocated according to expertise... • .. improving the quality of the final product • Different modules can be programmed in different languages • ... suitable for the application • ... good example • Reduces the amount of code that needs to be produced... • ... because code can be reused... • ... or standard library modules can be used... • ... reducing time of development • ... good example 	[4]	

	(b)	 <pre> graph TD A[Select songs] --> B[Enter criteria] A --> C[Perform search] A --> D[Display results] A --> E[Add song] C --> F[Search by title] C --> G[Search by artist] C --> H[Search by type] </pre> <p>* one mark each for each box under Select Song * one mark for the boxes under select song being in the right order * one mark for the three boxes under search the database * one mark for no incorrect lines</p>	[7]	
	(c)	Any 3 of: <ul style="list-style-type: none"> • A subroutine/subprogram/module of code... • ... which returns a single value • It can be called using its identifier • ... as part of an expression • The value returned replaces the function call 	[3]	
	(d)	<ul style="list-style-type: none"> • A data type which can accept only two values/(eg) TRUE or FALSE 	[1]	
	(e)	(i) <ul style="list-style-type: none"> • TRUE 	[1]	Accept 1, yes
		(ii) <ul style="list-style-type: none"> • FALSE 	[1]	Accept 0, no

Question	Expected Answers	Marks	Rationale
(f)	<p>Example</p> <pre> i = 1 WHILE i <= NumberOfSongs TotalLength = TotalLength + SongLength(i) i = i + 1 END WHILE </pre> <p>Several other solutions are possible. Award marks for the following mark points:</p> <ul style="list-style-type: none"> • i initialised correctly (typically 1 or 0) • A WHILE loop has been used correctly... • ... and the condition of the while loop (and the initial value of i) ensure that the correct number of iterations are made • Songlength(i) is added to the TotalLength <u>within each iteration</u> • The value of i is incremented (or decremented, as appropriate) <u>within each iteration</u> 	[5]	

Question			Expected Answers	Marks	Rationale
3	(a)	(i)	Any 3 of: <ul style="list-style-type: none"> • A data structure/contains several data items... • ... of the same data type... • ... grouped under one identifier • Individual items are accessed using an index • Stored contiguously in computer memory 	[3]	
		(ii)	Any 3 of: <ul style="list-style-type: none"> • Identifier/name • Data type • Dimensions (ie 1D, 2D etc..) • Size of the array/upper and lower bounds 	[3]	
	(b)		Correct answers only. In order these are: <ul style="list-style-type: none"> • 9 • 10 • Highest OR Column * 10 • 3 	[4]	

Question	Expected Answers	Marks	Rationale
(c)	<p>Example:</p> <pre> FOR Row = 1 TO 3 FOR Column = 1 TO 9 IF Ticket(Row, Column) = 0 THEN PRINT " " ELSE PRINT Ticket(Row, Column) END IF NEXT Column Go to new line NEXT ROW </pre> <p>Mark Points:</p> <ul style="list-style-type: none"> • Correct row loop • Correct column loop • Correctly nested/considers every element of the array Ticket • Checks if the value is 0 and ... • ... prints a space if it is 0 • ... prints the value if it is not 0 • There is a new line at the end of each row • Correct indentation has been used 	[8]	

Question	Expected Answers	Marks	Rationale
(d)	<p>High level response [6-8 marks] Candidates answer the question with complete and comprehensive explanations of many of the points below, justifying them by showing how they contribute to make the code more easily maintainable. Points made are supported by examples which may be linked to the application. The information will be presented in a structured and coherent form. There will be few, if any, errors in spelling, grammar and punctuation. Technical terms will be used appropriately and correctly.</p> <p>Medium level response [3-5 marks] Candidates answer the question with explanations of some of the points below. There may be some attempt to use examples or to link points to the application. The information will be presented in a structured format. There may be occasional errors in spelling, grammar and punctuation. Technical terms will be mainly correct.</p> <p>Low level response [0-2 marks] Candidates will demonstrate a limited understanding of the question. A few points from the list below will be made, but not clearly illustrated or argued. Information will be poorly expressed and there will be a limited, if any, use of technical terms. Errors of grammar, punctuation and spelling may be intrusive.</p> <p>Points to be made:</p> <ul style="list-style-type: none"> • Identifier names should describe the item identified... ... for example RowNumber, ColumnNumber ... so that the code can be read and understood without referring to a lookup table • Declarations should be made obvious within the code for examples by using comments with explanations of their purpose ... or separating them from code with blank lines ... so that they can easily be found, if you need to refer to them while reading the code • Standard conventions should be used 		

Question	Expected Answers	Marks	Rationale
	<p>... such as CamelCase/underscores/type prefixes ... as most translators do not allow spaces in identifier names/as a reminder of the data type of the variable or constant</p> <ul style="list-style-type: none"> • Keyword/reserved words in the language should be avoided ... eg you should not have a variable called Print/Count/Array/ other suitable example the translator will interpret this as the keyword and produce a syntax error • Declare constants and use these in the code instead of literals ... eg use NumberOfRows = 3, NumberOfColumns = 9 in this case ... code is easier to understand because the name of the constant rather than a literal is used (e.g for i = 1 to NumberOfRows) ... if the value changes, you only need to change it in the declaration of the constant (eg to print tickets with 4 or 5 rows) instead of looking for every instance of the literal • Variables/Constants should be declared as local wherever possible ... reducing the scope/lifetime of the variable to the minimum necessary ... avoiding errors due to clashing variable names in different parts of the program • Initialise variables when declared (in languages which allow this) ... e.g. int i = 1; string New = ""; ... this ensures that a suitable value is in the variable at the start of your algorithm 	[8]	

Question			Expected Answers	Marks	Rationale
4	(a)	(i)	<ul style="list-style-type: none"> • Takes a single character • ... and returns its ASCII code/character code in the computer's character set 	[2]	
		(ii)	Any 2 of: <ul style="list-style-type: none"> • The ASCII/Character codes for letters A-Z do not start from 1... • ...but start (in this case) from 65 / 65 is the ASCII code for 'A'... • ...and continue in sequence to 'Z' • Subtracting 64 corrects this offset. 	[2]	
	(b)		Any 3 of: <ul style="list-style-type: none"> • A function which calls itself • The original call is halted... • ... until subsequent calls return • Eventually reaches a stopping condition • For example the function mystery calls itself in line 55... • ... and the recursion will stop when $n < 10$ 	[3]	
	(c)		<ul style="list-style-type: none"> • Value returned : 5 Justification: <ul style="list-style-type: none"> • $n = 5$ • On line 52 the condition of the IF statement is TRUE/$n(5)$ is less than 10 • So on line 53, the value of n (which is 5) will be returned [1 mark for value returned, up to 2 for justification]	[3]	

Question	Expected Answers	Marks	Rationale
(d)	Any 5 of: Call Mystery(15)/n = 15 [No mark for this] <ul style="list-style-type: none"> • Line 52 – condition is FALSE • RETURN Mystery(15 – 9) (in line 55) • ... New call to Mystery(6)/n = 6 • ... Line 52 is TRUE • ... Return 6 to Mystery (15 - 9) (in line 55 of previous call) • Final return value is 6 	[5]	Candidates do not need to refer to line number to have the mark (but it must be clear what line of code they are on) <i>Allow follow through if the return value from second call is incorrect.</i>
(e)	<ul style="list-style-type: none"> • 9 (accept Temp >= 10) • Temp 	[2]	
(f)	Advantage Any 2 of: <ul style="list-style-type: none"> • Uses only one set of variables (which are updated on each loop)/ recursion creates new variables for each call • ... therefore more efficient use of memory... • ... and can be faster... • ... less likely to run out of stack space... Disadvantage Any 2 of: <ul style="list-style-type: none"> • Algorithm may be more difficult to follow/trace... • ... because variables are being reused • You need to be careful to get the conditions of loops correct (given current state of variables) • Humans often express the problem in a recursive way 	[4]	

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