OCR
Oxford Cambridge and RSA

GCE

Computer Science

Unit **H046/02:** Algorithms and problem solving

Advanced Subsidiary GCE

Mark Scheme for June 2016

OCR (Oxford Cambridge and RSA) is a leading UK awarding body, providing a wide range of qualifications to meet the needs of candidates of all ages and abilities. OCR qualifications include AS/A Levels, Diplomas, GCSEs, Cambridge Nationals, Cambridge Technicals, 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 2016

Annotations

Annotation	Meaning
^	Omission mark
BOD	Benefit of the doubt
	Incorrect point
FT	Follow through
NAQ	Not answered question
NBOD	No benefit of doubt given
REP	Repeat
	Correct point
TV	Too vague
BP	Blank Page – this annotation must be used on all blank pages within an answer booklet (structured or unstructured) and on each page of an additional object where there is no candidate response.
L1	Level 1
L2	Level 2
L3	Level 3

Question			Answer/Indicative content	Mark	Guidance

1			• Selection/Branching (1) (AO1.1)
			• Working selection example (1) (AO1.2)
			e.g.
			`if a>b then`
			`  c=b+42`
			`endif`
			
			• Iteration (1) (AO1.1)
			• Working iteration example (1) (AO1.2)
			e.g.
			`for count=1 to 10`
			`  print(count)`
			`next count`
			
			• Sequence (1) (AO1.1)
			• Working Sequence example (1) (AO1.2)
			e.g.
			`qty = input()`
			`total = qty * price`
				6
				AO1.1 (3)
				AO1.2 (3)
					Max 6 marks
					
					Do not penalise pseudocode if it is does not conform to the specification pseudocode guidelines.

Question			Answer/Indicative content	Mark	Guidance
2			**Mark Band 3–High Level** **(7-9 marks)** The candidate demonstrates thorough knowledge and understanding of reasons for the use of local and global variables and naming conventions; the material is generally accurate and detailed. The candidate is able to apply their knowledge and understanding directly and consistently to the context provided. Evidence/examples will be explicitly relevant to the explanation. The candidate provides a **thorough** discussion which is well-balanced (local/global **and** naming conventions). Evaluative comments are consistently relevant and well-considered. *There is a well-developed line of reasoning which is clear and logically structured. The information presented is relevant and substantiated.* **Mark Band 2-Mid Level** **(4-6 marks)** The candidate demonstrates reasonable knowledge and understanding of reasons for the use of local and global variables and naming conventions; the material is generally accurate but at times underdeveloped. The candidate is able to apply their knowledge and understanding directly to the context provided although one or two opportunities are missed. Evidence/examples are for the most part implicitly relevant to the explanation. The candidate provides a reasonable discussion, the majority of which is focused. Evaluative comments are for the most part appropriate, although one or two opportunities for development are missed.	**9*** **AO1.1 (2)** **AO1.2 (2)** **AO2.1 (2)** **AO3.3 (3)**	**AO1: Knowledge and Understanding** The following is indicative of possible factors/evidence that candidates may refer to but is not prescriptive or exhaustive: • Scope of global and local variables. Where declaration of global and local variables take place. • Duplication of variable name in separate functions • Variable identifiers must conform to a standard convention (meaningful name, camel back, data type indication, indicates global or local): this helps others to understand the code and reduces the likelihood of duplication, makes maintenance easier. • By convention UPPERCASE is reserved for constants rather than variables. • Programming languages have rules for names variables can have. Usually they can only contain letters, numbers and underscores and may not start with a number. Variable names cannot be reserved words (`if`, `while`, `for` etc). • Global variables make it difficult to integrate modules, they increase complexity of a program, they may cause conflicts with names written by others/in other modules, and they may be changed inadvertently when program is complex. • Local variables help to make each function reusable.

Question			Answer/Indicative content	Mark	Guidance
			There is a line of reasoning presented with some structure. The information presented is in the most part relevant and supported by some evidence. **Mark Band 1-Low Level** **(1-3 marks)** The candidate demonstrates a **basic** knowledge of reasons for the use of local and global variables and naming conventions, with **limited** understanding shown; the material is basic and contains some inaccuracies. The candidate makes a limited attempt to apply acquired knowledge and understanding to the context provided. The candidate provides a limited discussion which is narrow in focus. Judgments if made are weak and unsubstantiated. *The information is basic and communicated in an unstructured way. The information is supported by limited evidence and the relationship to the evidence may not be clear.* **0 marks** No attempt to answer the question or response is not worthy of credit.		**AO2.1: Application** The selected knowledge/examples should be directly related to the specific question. The following is indicative of possible factors/evidence that candidates may refer to but is not prescriptive or exhaustive: • Explanation of how the standard rules for programming would impact upon the choices made for using variables and how they are addressed. • Discussion of how breaking the rules of variable naming results in syntax error, causing it not to compile. • Discussion around the use of different variables that are dependent, independent or interdependent. **AO3.3: Evaluation** Candidates will need to consider a variety of viewpoints in relation to following standard rules for functions and variables while developing management software and will make evaluative comments about the issues and solutions they are discussing e.g. • Why when using global variables complexity of program increase? • Why meaningful variable names and camel back are needed? • Why indication of data type and whether local or global • Why local variables allow functions to be reusable? • Why variable rules are important so the tokeniser can recognise variables in lexical analysis. • Why ignoring rules of naming variables can result in unexpected behaviour (e.g. if a language is case sensitive `netPrice` is different to `netprice`)

<table>
<tr><th colspan="3">Question</th><th>Answer/Indicative content</th><th>Mark</th><th>Guidance</th></tr>
<tr><td>3</td><td>a</td><td></td><td>• Alpha
• Beta
• Blackbox</td><td>3
AO1.1
(3)</td><td>Max 3 marks
Accept Integration testing, Unit testing, Destruction testing, walkthrough/dry-run, acceptance</td></tr>
<tr><td></td><td>b</td><td></td><td>• the (complete) structure / logic has been tested. (1)
• because it is safety critical (1) e.g. chemical leak (1)
• potential impact on life (1)</td><td>3
AO2.2
(3)</td><td>Max 3 marks</td></tr>
<tr><td></td><td>c</td><td></td><td>
<table>
<tr><th>Sensor Value</th><th>Output</th><th>On/Off</th></tr>
<tr><td rowspan="2">95.000</td><td>Heater</td><td>On</td></tr>
<tr><td>Amber Light</td><td>Off</td></tr>
<tr><td rowspan="2">97.500</td><td>Heater</td><td>Off</td></tr>
<tr><td>Amber Light</td><td>Off</td></tr>
<tr><td>97.501</td><td>Amber Light</td><td>On</td></tr>
<tr><td rowspan="2">98.099</td><td>Amber Light</td><td>On</td></tr>
<tr><td>Red light</td><td>Off</td></tr>
<tr><td rowspan="2">98.100</td><td>Amber Light</td><td>Off</td></tr>
<tr><td>Red Light</td><td>On</td></tr>
</table>
</td><td>5
AO2.2
(5)</td><td>One mark per band
Order within each band unimportant

Max 5 marks</td></tr>
<tr><td>4</td><td>a</td><td></td><td>• Find the middle point in the list / 21 / element 4
• Compare it to the value 47, false
• Is 47 greater than middle point, true
New subset is 46-51 / change lower bound to 46 / element 5
• Find the middle of the new subset / 47 / element 6
Is this value equal to 47, true Search finishes</td><td>4
AO2.1

(4)</td><td>Some marks such as the comparison may be by implication if the candidate’s logic works

Must refer to the list given in the question i.e. not a generic description</td></tr>
</table>

<table>
<tr><th colspan="3">Question</th><th>Answer/Indicative content</th><th>Mark</th><th>Guidance</th></tr>
<tr><td></td><td>b</td><td></td><td>

- Finding `midpoint` and correctly checking if `midPoint` value is target value …
- … and if so returning `true`
- Correctly checking that all elements have been checked …
- … and if so returning `false`
- Identify top or bottom of list …
- … if top then `leftPtr` set/passed as `midPoint + 1` …
- … if bottom then `rightPtr` set/passed as `midPoint - 1`
- Correct use of indentation (AO2.1)

</td><td>8
AO3.2 (7)

AO2.1 (1)</td><td>Max 8 marks

Note: candidates may have given a recursive algorithm and this should is perfectly acceptable.</td></tr>
<tr><td colspan="6">

Example iterative example

```
function findItem (numberArray integer[2000], targetNumber:integer, leftPtr:integer,
rightPtr:integer): boolean

  while (leftPtr <= rightPtr)

      midPoint = (leftPtr + rightPtr) DIV 2
      if (numberArray[midPoint] == targetNumber)
        return true
      else if (numberArray[midPoint] < targetNumber)
        leftPtr = midPoint + 1
      else
        rightPtr = midPoint - 1
      endif

  endwhile
  return false
endfunction
```

</td></tr>
</table>

Question			Answer/Indicative content	Mark	Guidance
	c	i	• The integers in the list are unsorted (1)	**1 AO2.1 (1)**	
		ii	Identification (Max 1) • Perform a linear search Description (Max 2) • starting at the first element / each item is checked… • until value is found • or end of list reached and not found	**3 AO1.1**	Accept serial

Question			Answer/Indicative content	Mark	Guidance

	d		• String length calculated (1)
			• Correct number of characters from passed string taken … (1)
			• … in reverse order (1)
			• Characters placed in stack in correct order (1)
			• String length placed in stack at correct point (1)
			• Meaningful variable names used (1) (AO2.1)
			
			Example program
			
			`procedure passToStack(passString)`
			`  stringLen = passString.Length()`
			`  if stringLen == 0 then`
			`  stack[0]=0`
			`  else`
			`  stackPtr = 0`
			`  stringPtr = stringLen - 1`
			`  for i = 1 TO stringLen`
			`  stack[stackPtr] = passString[stringPtr]`
			`  stackPtr = stackPtr + 1`
			`  stringPtr = stringPtr -1`
			`  next i`
			`  stack[stackPtr] = stringLen`
			`  endif`
			`endprocedure`
				6
				AO3.2 (5)
				
				AO2.1 (1)
					Allow `StackPtr` to be used instead of `i` in loop, as we would not expect them to know that some compilers do not always increment "loop counter" when they exit loops (i.e. loop counter on exit is undefined)
					
					Accept candidates using built-in stack methods
					e.g. `stack.push(word.substring(i,1))`
					
					Do not penalise for syntax errors if the logic can clearly be followed.
					
					Max 6 mark

Question			Answer/Indicative content	Mark	Guidance
5	a		e.g. • Reduces track scenery • Limited functionality on car dashboard • Simplified controls • Simplified physics • Simplified / removed weather	**2 AO2.1 (2)**	Accept any reasonable answer
	b		e.g. • Track name • Car type • Driver name	**3 AO2.1 (3)**	Allow any reasonable alternative

6	a)		• Read in `A` and `B`.
			• Correct comparisons
			• Correct output messages.
			• Open file
			• Write to and close file.
			
			E.g.
			
			`A = input("Enter value A")`
			`B = input("Enter value B")`
			`myFile = openWrite("output.txt")`
			`if A < B then`
			`  myFile.writeLine("A is less than B")`
			`elseif B < A then`
			`  myFile.writeLine("B is less than A")`
			`else`
			`  myFile.writeLine("A is equal to B")`
			`endif`
			`myFile.close()`
				5 AO3.1 (5)	Max 5 marks
					
					Accept open file in append mode

Question			Answer/Indicative content	Mark	Guidance
6	b		Parameter / name is passed by value… …rather than by reference / by value does not change the original variable value		
6	c)		• Work is easier to divide between a team • each team member just needs to know what values go into their subroutine and the expected functionality • Saves time as work takes place in parallel • each team member can work on their area of expertise. • Breaks problems into smaller areas. • Easier to test/ debug/ read • each subroutine can be tested before integration. • Code can be reused in the project/ future projects	**6** **AO1.1**	Maximum 6 marks
7		a	• 16	**4** **AO2.2** **(4)**	Max 4 marks
		b	• 16		
		c	• 15		
		d	• 42 • 43 (max. 1)		

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

OCR Customer Contact Centre

Education and Learning
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

© OCR 2016

PART OF THE CAMBRIDGE ASSESSMENT GROUP

LLOYD'S REGISTER QUALITY ASSURANCE · ISO9001 · UKAS MANAGEMENT SYSTEMS 001