OCR
Oxford Cambridge and RSA

GCE

Computer Science

Unit **H446A/01:** Computer systems

Advanced GCE

Mark Scheme for June 2017

H446/01 **Mark Scheme**

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.

These are the annotations, (including abbreviations), including those used in scoris, which are used when marking

Annotation	Meaning
^	Omission mark
BOD	Benefit of the doubt
C	Subordinate clause / consequential error
	Incorrect point
E	Expansion of a point
FT	Follow through
NAQ	Not answered question
NBOD	No benefit of doubt given
P	Point being made
REP	Repeat
	Correct point
TV	Too vague
0	Zero (big)
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	Marks	Guidance
1	a	i	To render models of proposed buildings. (1) Run CAD software. (1) Run modelling calculations. (1) Any example sensible to scenario. (1) (Max 1)	1 (AO2.1)	
		ii	Random Access Memory (1) A form of primary memory (1) Used to hold data and/or programs <u>in use</u> (1) Volatile/Loses its contents when power is lost. (1) (Max 2)	2 (AO1.1)	
		iii	Multiple Cores (1) High/Fast Clock Speed (1) Ability to use pipelining (1) Large Cache (1) (Max 1)	1 (AO1.1)	Accept concurrency/parallel processing for pipelining
	b	i	Paging…(1) …Memory is divided into fixed/physical units(1) Segmentation… (1) …Memory is divided logically/variable size according to its contents. (1)	4 (AO1.1)	Accept same size units for MP1
		ii	Multitasking allows the user to run more than one program <u>at the same time</u>. (1) E.g. running CAD software whilst checking emails. (1)	2 (AO1.1 – 1 mark AO1.2 – 1 mark)	Accept any reasonable work related answer

	c	i	Advantage: - Centrally administered in one location. (1) - One location to back up. (1) Disadvantage: - Central point of failure. (1) - Can be expensive to maintain/set up (e.g. cabling costs, specialist staff.) (1) (Max 1 Advantage, 1 Disadvantage)	2 (AO1.2)	Accept for MP1 better security Do not credit quick access as an advantage
		ii	A hardware device/piece of software that monitors (and filters/blocks) traffic/packets going to and from a network. (1) (Max 1)	1 (AO1.1)	Accept 'content' for 'traffic/packages'
		iii	Prevent unauthorised access to a network. (1) To restrict applications that are used internally that have internet access. (1) To restrict websites that can be accessed from within the company. (1) To protect the company's data/intellectual property. (1) (Max 1)	1 (AO 1.2)	Accept for MP1 malicious attacks/traffic
2	a	i	A dynamic/data structure (1) Each node/item consists of data and pointer (1) Pointer gives location of next node. (1)	3 (AO1.2)	Accept 'element' instead of 'node/item'

		ii	Description can be written: - Oxford pointer changed to bypass Birmingham and point to Manchester. (1) - A node is created holding the data York/York is placed is next free space/node/item (1) - Manchester remains in original position and pointer changed to point to the York node. (1) - The York node points to null (or terminator). (1) OR via diagram eg.:  	4 (AO2.1)	On diagram don't penalise if the pointer from Birmingham is left intact. It should be clear in both diagram and text that Oxford no longer points to Birmingham. In diagram solution, London, Oxford and Manchester must remain in the same positions.
	b		A linked list requires every node to be checked (until the desired record is found). (1) A linked list will take longer to search (as more nodes are added). (1) A hash table enables direct access to the location of the record. (1) A hash table will take the same time to search (as more nodes are added)/It takes no longer as more records are added. (1)	4 (AO1.2 - 2 marks AO2.2 - 2 marks)	Some candidates may talk about time complexity: linked lists being linear/O(n) and hash table being constant/O(1) Accept these as points 1& 2 and 3 & 4 conjoined i.e. full marks.
3	a	i	Downloads quicker. (1) Saves user money by using less bandwidth/ on data usage. (1) (Max 1)	1 (AO1.2)	Do not accept 'saves the user space on their device'.

		Ii	Lossy takes away some of the information from the original. (1) Lossless preserves all the information from the original. (1) With text the loss of small amounts of information will make it unreadable. (1)	3 (AO1.1 – 2 marks AO2.1 - 1mark)	

	b		**Mark Band 3–High Level (9-12 marks)** The candidate demonstrates a thorough knowledge and understanding of dictionary and run length encoding for compression. 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 is able to weigh up both forms of compression and justify dictionary encoding being the better choice. 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 (5-8 marks)** The candidate demonstrates reasonable knowledge and understanding of dictionary and run length encoding for compression; 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 makes a reasonable attempt to come to a conclusion as to which form of compression is better	AO1.1 (2) AO1.2 (2) AO2.1 (3) AO3.3 (5) 12	Points may include but aren't limited to: **AO1 Knowledge and Understanding** Run length encoding relies on consecutive pieces of data/characters being the same. Each set of consecutive symbols can be represented by the symbol and its number of occurrences e.g. AAAABBBBBCCC could be represented as 4A5B3C (or A4B5C3 or any sensible RLE encoding) In dictionary encoding frequently occurring pieces of data/groups of characters are replaced by symbols/tokens/smaller groups of characters/indexes. A dictionary is then used to say which symbols/tokens/characters/indexes match which groups of characters. When decompressed the dictionary is used to replace the tokens with the original text. **AO2.1 Application** Run Length Encoding is very unsuitable for the example text There are very few consecutive repeating symbols in the text. only instances being ll and ee these still require 2 characters to represent them 2l and 2e Dictionary encoding is well suited. There are lots of repeating groups of characters For example 'call' 'name' '[SPACE]we' 'Romeo' We could for example have: What's in53? that which2 15 rose

			suited. 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 (1-4 marks)** The candidate demonstrates a basic knowledge of dictionary and run length encoding for compression; 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 nothing more than an unsupported assertion. **0 marks** No attempt to answer the question or response is not worthy of credit.		By5ny other3 would smell5s sweet; So4would,2re he not41'd 1:call 2:[space]we 3:[space]name 4:[space]Romeo[space] 5:[space]a (NB candidates are unlikely to show full compression, just a demonstration of the principle is sufficient. The best candidates are likely to show an awareness that space is a character that can be used in compression and that upper and lowercase letters are different. Demonstrating this is indicative of but not a requisite of the band.) **AO3.3: Evaluation** Run length encoding is not suited to natural language (more likely to be used in simple images). Applying it to the example the resulting text would be the same size as the original/worse than the original (if we use 1s to represent every individual instance of a character). Dictionary encoding works well. We can already see benefit on small piece of text. Would fare substantially better on full works. Dictionary encoding is the best compression method for this scenario.

<table>
<tr>
<td>4</td>
<td>a</td>
<td></td>
<td>Q ≡ (A∧B) ∨ (C∧D)

1 mark for (A∧B)

1 mark for (C∧D)

1 mark for the ∨ joining the two parts.</td>
<td>3

(AO1.2)</td>
<td>Accept (C∧D) ∨ (A∧B)

Accept (B∧A) instead of (A∧B)

Accept (D∧C) instead of (C∧D)

Accept alternative notations (e.g. +/. OR/AND)

Accept AB as (A.B) and CD as (C.D)

Accept answers without brackets</td>
</tr>
<tr>
<td></td>
<td>b</td>
<td>i</td>
<td>
<table>
<tr><th>E</th><th>F</th><th>G</th><th>(E∧F)</th><th>(E∧G)</th><th>(E∧F)∨(E∧G)</th></tr>
<tr><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
</table>
1 mark for each of the pairs of rows.
</td>
<td>4

(AO1.2)</td>
<td></td>
</tr>
</table>

<table>
<tr><td></td><td></td><td>ii</td><td>(F∨G) ∧ E

One mark for the (F∨G)
One mark for the ∧ E</td><td>2

(AO2.2)</td><td>Accept:
(G∨F) ∧ E
E∧(F∨G)
E∧(G∨F)</td></tr>
<tr><td>4</td><td>c</td><td></td><td>Encrypt the film (1)
Send the key/password out on the release date (1)</td><td>2

(AO2.2)</td><td>Accept
Use Digital Rights Management/DRM…
..To keep content encrypted until given date.</td></tr>
<tr><td>5</td><td>a</td><td>i</td><td>- Stores the value 10 (1)
- In a memory location (1)
- Given the label/symbolic address ten (1)</td><td>3

(AO1.2)</td><td>MP3 Accept identifier</td></tr>
<tr><td></td><td></td><td>ii</td><td><table><tr><th>Starting value in Accumulator</th><th>Pass or Fail</th></tr><tr><td>29</td><td>Fail</td></tr><tr><td>30</td><td>Pass</td></tr><tr><td>31</td><td>Fail</td></tr></table>1 Mark per row</td><td>3

(AO2.1)</td><td></td></tr>
<tr><td></td><td>b</td><td>i</td><td>LDA (1)
SUB (1)
ADD (1)
INP (1)
(Max 1)</td><td>1

(AO1.2)</td><td></td></tr>
</table>

		ii	BRA (1) BRP (1) BRZ (1) (Max 1)	1 (AO1.2)	
		iii	20	1 (AO2.1)	
		iv	40	1 (AO2.1)	
		v	Rounds up (the number input)… (1) …To the nearest multiple of ten (and outputs it) (1)	2 (AO2.2)	Rounds to multiple of ten gets one mark.
6	a	i	10111100	1 (AO1.2)	
		ii	BC	1 (AO1.2)	
	b	i	10101100	1 (AO1.2)	

		ii	11010100	1 (AO1.2)	
	c		Shift Right (1) Two Places (1)	2 (AO1.2)	Allow one mark for correct number of places but wrong direction.
	d		Binary point: shifted four places gives: 01001.0 (1) Binary point shifted two places gives: 010.010 (1) Subtraction carried out … 01001.000 – 010.010 (1) …'Borrowing' shown… (1) …Answer: 0110.110 (1) Normalised to: Mantissa 011011 (1) Exponent 0011 (1)	6 (AO1.2)	Correct answer with clear binary subtraction/2's complement addition calculation gives full marks.
7	a		A field which has a unique value for every record/A unique identifier. (1)	2 (AO1.1 –	

			E.g. userID (1)	1 , AO2.1 -1)	
	b	i	A result generated by applying an algorithm/numeric process to a value. (1)	1 (AO1.1)	
		ii	Hash functions are one way/can't be reverse (1) If someone gains access to the database they cannot access user's password. (1)	2 (AO1.2 1 mark, AO2.1 1 mark)	
	c		`SELECT passwordHash, locked` (1) `FROM Users` (1) `WHERE username='Apollo'` (1)	3 (AO 3.2)	Do not award first mark for SELECT *

	d		`UPDATE Users` (1)
			`SET locked=1`(1)
			`WHERE username='Hades'` (1)
				3
				(AO 3.2)
					Allow other updating method e.g. a DELETE statement followed by an INSERT statement, for full marks e.g.
					`DELETE FROM Users`
					`WHERE username = 'Hades'` (1 mark)
					`INSERT INTO Users` (1 mark)
					`VALUES (<userID value>,'Hades',<passwordHash value>,1)`
					(1 mark)

	e		Takes a hash of `givenPassword` (NB this may be done inline e.g. `if hash(givenPassword)==passwordHash and locked==0 then` (1)
			
			Returns true if password is correct and account is unlocked. (1)
			
			Returns false if account is locked (1)
			
			Returns false if password is incorrect (1)
				4
				
				(AO 3.2)
					Example code:
					`temp = hash(givenPassword)`
					`if temp==passwordHash and locked==0 then`
					`  return true`
					`else`
					`  return false`
					`endif`
					
					Candidates may have taken a different approach – any solution that fulfils the criteria on the left should get them marks.

8			**Mark Band 3–High Level (7-9 marks)** The candidate demonstrates a thorough knowledge and understanding issues around computers and the workforce and Artificial Intelligence. 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. 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	AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3) 9	Points may include but aren't limited to: **AO1 Knowledge and Understanding** Artificial Intelligence (AI) is the study of computers displaying intelligent behaviour (usually characterised by decision making). AI techniques include neural networks, evolutionary computation, Bayesian networks etc. Computers are well suited to certain jobs and as AI techniques improve the range of jobs they can do is likely to increase. **AO2.1 Application** **For the argument:** Many jobs have already been taken over by computers. Manual job such as work in the automotive industry has been replaced by robots.

			knowledge and understanding issues around computers and the workforce and Artificial Intelligence; 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 sound discussion, the majority of which is focused. Evaluative comments are for the most part appropriate, although one or two opportunities for development are missed. 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 around computers and the workforce and Artificial Intelligence.; 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		Computer based systems are attractive to employers, they don't require paying, don't get sick and can work 24/7 without making mistakes. Computer systems can be used for work that is considered dangerous for humans And for repetitive and menial tasks. Future developments may make computers better at highly skilled tasks making computers preferable And in the case of certain tasks (e.g. surgery) would make the use of human workers unethical. May cite recent developments in AI (e.g. beating world Go Champion) **Against the argument:** As technology develops people will be required to design these new systems. It is likely to assist but not take over all roles Producing a more skilled workforce Developments in AI have been forecast for many decades but never materialised. Forecasts as to the development of AI have always been wildly optimistic There is debate as to whether AI will ever be able to show human levels of intelligence. Tasks that humans find inherently 'easy' are still beyond the reach of computers. Crucially for many areas of work computers will have to pass the Turing Test – i.e. converse to such a level that will enable them to pass as

			worthy of credit.		human… …Most believe this is still well beyond our current understanding. Laws would need changing as currently people have the right to have automated decisions checked over by humans (DPA) **AO3.3 Evaluation** Candidate should have come to a well reasoned conclusion for or against the argument. They could come down on either side, the important thing is they have considered both points of view and based their conclusion on the evidence they have discussed.
9	a		Only one element can have a given id/id is unique. (1) Class can be used assigned to multiple elements/used multiple times. (1)	2 (AO1.1)	
	b		`h1{` (1 mark for open and close) `font-family:Arial(;)` (1 mark) `}` `.customerQuote{` (1 mark) `background-color: #E8C3E1(;)` (1 mark) `}` `#intro{` (1 mark) `(font-)color: darkRed(;)` (1 mark) `}`	6 (AO3.1)	`.customerQuote` must have . and opening and closing { } for 3rd mark. `#intro` must have # and opening and closing { } for 4th mark Must match case sensitivity, except for 'Arial' and 'darkRed' and colour code Allow quotes around `Arial` and `darkRed`

	c	i	JavaScript	1 (AO1.1)	**Cao** do not accept Java
	ii		Change line `if(hour>9 && hour<17)` … … To `if(hour>8 && hour<17)` (1 Mark) or … `if(hour>=9 && hour<17)` (1 Mark)	1 (AO3.3)	Accept Change 'greater than' to 'great than or equal to' or similar
	iii		- Won't work if JavaScript is disabled. (1) - Shows incorrect message if user's computer's clock is wrong/in different time zone. (1) - (Source) code is visible allowing it to be copied/modified. (1) (Max 1)	1 (AO2.2)	
10	a		- Prototype is created (1) - (Evaluated and) feedback used to inform next iteration (1) - Any changes are made (1) - Process repeated until…(1) … prototype becomes final product. (1) (Max 4)	4 (AO1.1)	

	b		- Function traverses every letter of answer (1)
			- Function traverses every randomLetters (1)
			- Correctly checks each letter of answer against each of randomLetters (1)
			- Returns 0 if answer contains a letter that doesn't occur in randomLetters (1)
			- Returns 0 if letter occurs more times in answer than randomLetters (1)
			- Returns answer length for a valid word.(1)
				6
				(AO3.2)
					i=0
					while i<answer.length
					j=0
					letter=answer.substring(i,1)
					while j<10 and randomLetters[j]!=letter
					j=j+1
					endwhile
					if j<10 then
					randomLetters[j]="!"
					else
					return 0
					endif
					i=i+1
					endwhile
					return answer.length

	c		BS Tree can be searched quicker than an array.	1 (AO1.2)	Accept O(log n) search time rather than O(n)

	di		Saves time/money as prewritten (1) Draws on expertise of other programmers (1) Pre-tested (so likely to work) (1) Can have been written in a different language (1) (Max 2)	2 (AO1.2)	
	dii		**Mark Band 3–High Level (7-9 marks)** The candidate demonstrates a thorough knowledge and understanding of how source code is compiled and library code incorporated. 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. 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 how source code is compiled and library code incorporated; the material is generally accurate but at times underdeveloped. The candidate is able to apply their knowledge and understanding directly to the context	AO1.1 (2) AO1.2 (2) AO2.1 (2) AO3.3 (3) 9	Points may include but are not limited to: **AO1 Knowledge and Understanding** The compiler is effectively a group of programs. The stages of compilation are: lexical analysis, syntax analysis, code generation and optimisation. A linker is then used to combine the object code with the library code to make the final executable. **AO2.1 Application** Source code is input into a compiler program. The first stage is lexical analysis in which.. Comments and whitespace are removed Variables, and subroutines stored in symbol table Which also holds data such as scope and data type Code is converted to a series of tokens The series of tokens and symbol table is passed onto the next stage, syntax

provided although one or two opportunities are missed. Evidence/examples are for the most part implicitly relevant to the explanation.

The candidate provides a sound discussion, the majority of which is focused. Evaluative comments are for the most part appropriate, although one or two opportunities for development are missed.

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 how source code is compiled and/or library code incorporated; 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.

analysis:

Here the code is checked to ensure it follows the rules of the language.

This is often accomplished by placing the tokens into a (abstract syntax) tree.

Where it breaks the rules of the language errors are generated.

If no rules are broken then it's passed on to the next stage…

..Which is code generation.

Here the object code (accept machine code) is created.

(i.e. the binary that is executed by the processor)

This code may be inefficient..

.. it may contain unnecessary instructions or groups of instructions that can be replaced by simpler ones.

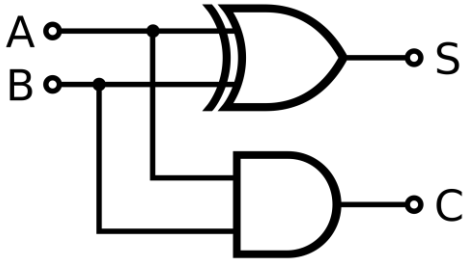
Code from the library is likely already compiled.

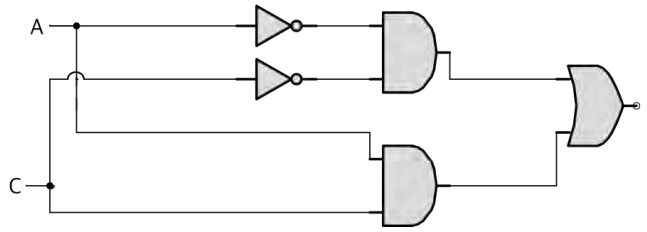
And may well have been written in a different language to the main program.

The main program source code will have contained lines importing the library code.

A program called a linker can incorporate the code from the library with the main program…
…into a single executable file.
An alternative approach is for the main executable to link to the compiled library code (i.e. dynamic linking).

AO3.3 Evaluation
Lexical analysis is necessary to put the code into a format which can be

					read and processed (i.e. parsed) by the syntax analyser. Syntax Analysis is necessary to ensure the code is valid in as much as it meets all the structural rules of the language. This guarantees it will run (though it might not do as expected and may still have occurrences of run-time errors). Code generation is necessary to turn the code into a format that the processor can understand (i.e. binary machine code). The code optimisation whilst not necessary, does ensure the code runs quicker or using less memory. Linking is necessary to ensure the library code is incorporated into the final program.
11	a		 XOR Gate (1) AND Gate (1) Correct connections and no additional gates (1)	3 (AO1,1)	

11	b		- Correctly identified groups on Karnaugh map/Correct boolean statement.(1) - NOT A AND NOT C Gates (1) - A AND C gates (1) - Both sets of gates joined by OR gate (with no other gates used). (1)	4 (AO2.2)	**AB** 00 01 11 10 **CD** 00: 1 1 0 0 01: 1 1 0 0 11: 0 0 1 1 10: 0 0 1 1 (¬A ∧ ¬C) ∨ (A∧C) Or equivalent.  Or equivalent.

Question	Assessment Objectives							Total
	AO1.1	AO1.2	AO2.1	AO2.2	AO3.1	AO3.2	AO3.3	
1ai			1					1
1aii	2							2
1aiii	1							1
1bi	4							4
1bii	1	1						2
1ci		2						2
1cii	1							1
1ciii		1						1
2ai		3						3
2aii			4					4
2b		2		2				4
3ai		1						1
3aii	2		1					3
3b*	2	2	3				5	12
4a*m*		3						3
4bi*m*		4						4
4bii*m*				2				2
4c				2				2
5ai		3						3
5aii			3					3
5bi		1						1
5bii		1						1
5biii			1					1
5biv			1					1
5biv				2				2
6ai*m*		1						1
6aii*m*		1						1
6bi*m*		1						1
6bii*m*		1						1
6c*m*		2						2
6d*m*		6						6
7a	1		1					2
7bi	1							1
7bii		1	1					2
7c						3		3
7d						3		3
7e						4		4
8*	2	2	2				3	9
9a	2							2
9b					6			6
9ci	1							1

9cii							1	1
9ciii				1				1
10a	4							4
10b						6		6
10c		1						1
10di		2						2
10dii*	2	2	2				3	9
11a*m*	3							3
11b*m*				4				4
	29	**44**	**20**	**13**	**6**	**16**	**12**	140

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