Version 1.0 0712

General Certificate of Education (A-level)
June 2012

Computing COMP3

(Specification 2510)

Unit 3: Problem Solving, Programming, Operating Systems, Databases and Networking

Final

Mark Scheme

Mark schemes are prepared by the Principal Examiner and considered, together with the relevant questions, by a panel of subject teachers. This mark scheme includes any amendments made at the standardisation events which all examiners participate in and is the scheme which was used by them in this examination. The standardisation process ensures that the mark scheme covers the candidates' responses to questions and that every examiner understands and applies it in the same correct way. As preparation for standardisation each examiner analyses a number of candidates' scripts: alternative answers not already covered by the mark scheme are discussed and legislated for. If, after the standardisation process, examiners encounter unusual answers which have not been raised they are required to refer these to the Principal Examiner.

It must be stressed that a mark scheme is a working document, in many cases further developed and expanded on the basis of candidates' reactions to a particular paper. Assumptions about future mark schemes on the basis of one year's document should be avoided; whilst the guiding principles of assessment remain constant, details will change, depending on the content of a particular examination paper.

Further copies of this Mark Scheme are available from: aqa.org.uk

Set and published by the Assessment and Qualifications Alliance.

The Assessment and Qualifications Alliance (AQA) is a company limited by guarantee registered in England and Wales (company number 3644723) and a registered charity (registered charity number 1073334).
Registered address: AQA, Devas Street, Manchester M15 6EX.

COMPONENT NUMBER: COMP3

COMPONENT NAME: Problem Solving, Programming, Operating Systems, Databases and Networking

STATUS: Final

DATE: July 2012

To Examiners:

1. **When to award '0' (zero) when inputting marks on CMI+**
   A mark of 0 should be awarded where a candidate has attempted a question but failed to write anything credit worthy.

   Insert a hyphen when a candidate has not attempted a question, so that eventually the Principal Examiner will be able to distinguish between the two (unattempted/nothing credit worthy) in any statistics.

2. This mark scheme contains the correct responses which we believe that candidates are most likely to give. Other valid responses are possible to some questions and should be credited. Examiners should refer off mark scheme responses that they believe are creditworthy to a Team Leader.

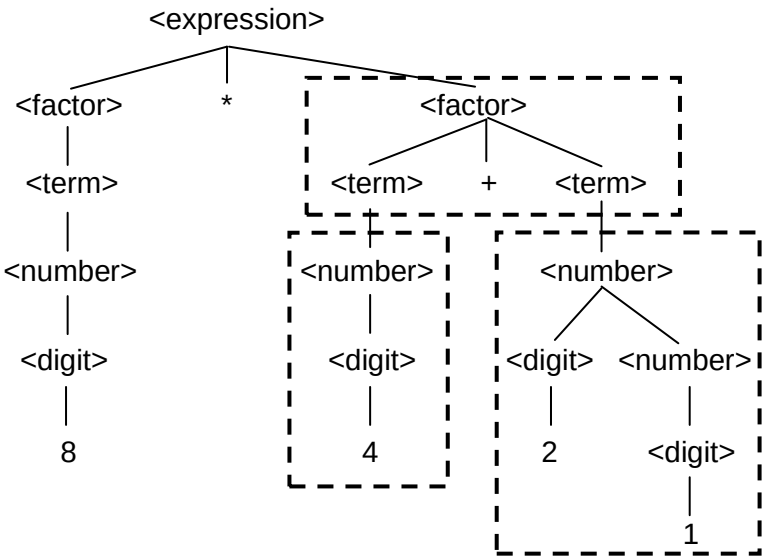
The following annotation is used in the mark scheme.

; - means a single mark
// - means alternative response
/ - means an alternative word or sub-phrase
A - means acceptable creditworthy answer
R - means reject answer as not creditworthy
NE - means not enough
I - means ignore
DPT - in some questions a specific error made by a candidate, if repeated, could result in the loss of more than one mark. The **DPT** label indicates that this mistake should only result in a candidate losing one mark, on the first occasion that the error is made. Provided that the answer remains understandable, subsequent marks should be awarded as if the error was not being repeated.

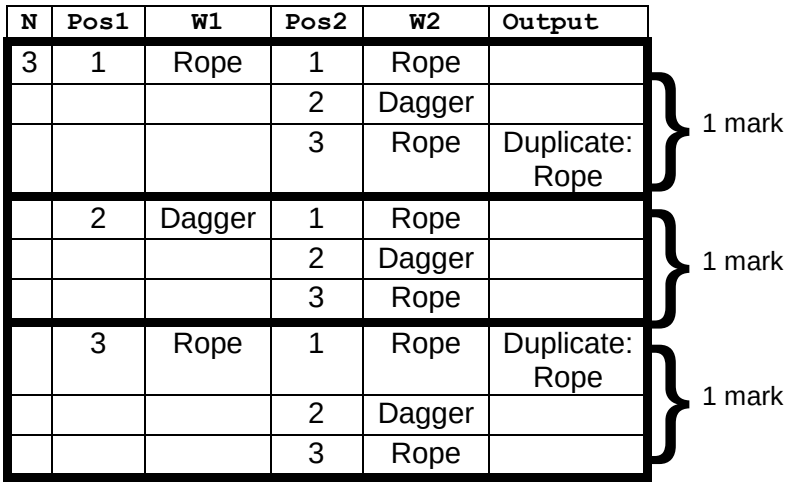
1			Processor management // Allocation of processors // Allocation of processor time // (process) scheduling // thread management; **A** processing management Allocation/management of RAM / memory // allocation of buffers; Allocation/management of / control of I/O devices/peripherals // I/O management // device driver management; File / backing store / secondary store management / access / organisation; Power / battery management; **A** Interrupt handling; **A** Provision of Application Program Interface / API; **A** interface between hardware and applications **A** Provision / management of (windows in) user interface; **A** Management of system security; **A** Answers by example, only one example of each type **A** A description of a type of software management but not just "software management". e.g. loading of programs, software installation, registering DLLs. **R** Software management alone unless role of OS in this is clear e.g. installation of new software, updating registry **MAX 3**	3

2	(a)		Backus-Naur (Form); **A** Backus Normal (Form), BNF, Extended Backus-Naur (Form), Augmented Backus-Naur (Form), ABNF **A** Misspellings of Backus-Naur **A** Format for Form and the word "Notation" **R** BN	1

2	(b)		<table><tr><th>Statement Type</th><th>String</th><th>Valid (Yes/No)</th></tr><tr><td><number></td><td>129.376</td><td>No;</td></tr><tr><td><factor></td><td>23 + 17</td><td>Yes;</td></tr></table> **A** Alternative clear indicators of Yes or No e.g. Y/N, Tick/Cross, Valid/Invalid, True/False	2

2	(c)		 **1 mark** for each area surrounded by a rectangle **A** missing chevrons **DPT** Arrows drawn instead of lines	3

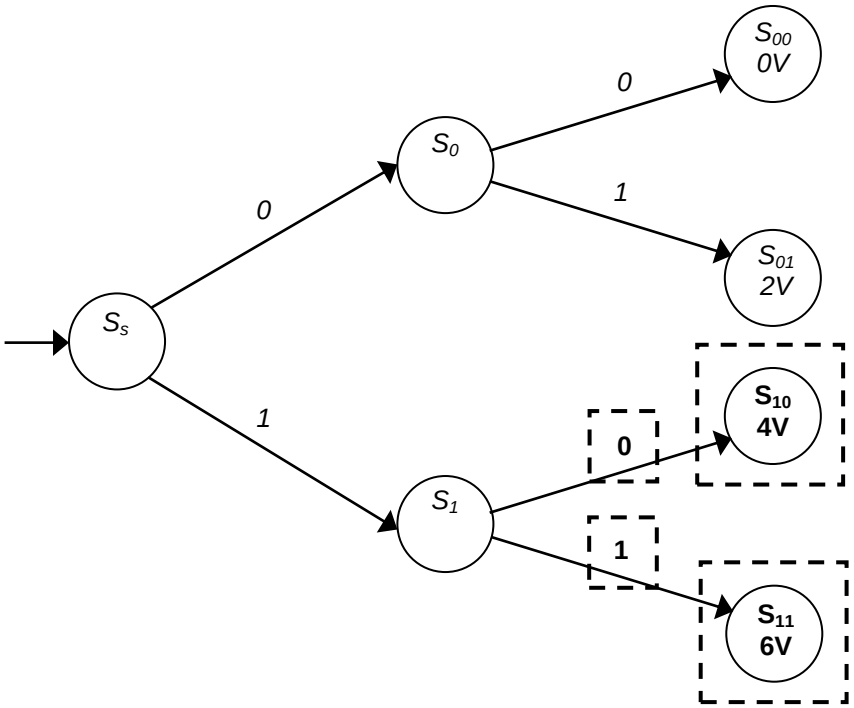
3	(a)		Space / Memory (complexity); **A** amount of memory used	1

3	(b)	(i)	 **A** answers which have correct values repeated in empty cells, but do not award a mark if there are any incorrect values within the block for which the mark is being awarded. **A** additional rows in trace table, so long as the trace is correct. **DPT** if just "Duplicate" or "Rope" are written in the Output column when it should be "Duplicate: Rope" or if the value of Pos1 is written in the output instead of W1 e.g. 1 instead of "Rope" If candidate has not written in the value of N, only one mark should be lost (for the top rectangular area) for this mistake	3
3	(b)	(ii)	$O(n^2)$;	1
3	(b)	(iii)	**MARKS CAN ONLY BE AWARDED IF CORRECT ANSWER FOR PART 3 (b) (ii)** **Alternative 1:** Algorithm has nested loops // two loops with one inside the other; **A** reference to inner and outer loops Each loop repeats N times; **Alternative 2:** The (basic) operation / If statement / file read / comparison is carried out N^2 times; because it is inside nested loops // because each loop executes N times; **Alternative 3:** Each of the (N) entries is compared to each of the (N) others // each entry is compared N times; so N^2 (**A** N*N) comparisons/operations are required // $N*N=N^2$; **A** uppercase or lowercase n **A** answers where examples are used instead of N and N^2, e.g. 3 and 9. **A** check as alternative to comparison **MAX 2**	2

4	(a)		Greater the bandwidth, the higher the bit rate // positive correlation // (directly) proportional; Bandwidth must be at least 2wHz where w is the bit rate in bits per second;	MAX 1

4	(b)		Time delay between the moment something is initiated and the moment its effect begins **A** time delay between signal being transmitted and arriving **A** time taken for transmitted data to arrive at the receiver **A** lag for time delay **NE** delay in transmission, transmission time	1

4	(c)		Bit rate is double/twice baud rate // Baud rate is half bit rate; **A** "It" is double; **A** 2:1	1

4	(d)		 **1 mark** for labelling a transition arrow with 0 **1 mark** for labelling a transition arrow with 1 **1 mark** for labelling a state with the value 4V and a unique state name **1 mark** for labelling a state with the value 6V and a unique state name **MAX 2 if the states and transition arrow labels do not correspond** Note that: • The state names do not have to match those given here. • The voltage values can be followed by a V, the word Volts or nothing. • The zero and one on the transition arrows to the right of S_1 can be either way around e.g. 1 above 0 is okay.	4

5	(a)		(Using an algorithm) to convert a message into a form that is not understandable (without the key to decrypt it); (Using an algorithm) to convert a message into a form that is only understandable by the intended parties // can only be read with the correct key; Converting a message into cipher text; **NE** scrambling unless further explanation is provided **A** "unreadable" for "understandable" **A** "data" for "a message" **MAX 1**	1

5	(b)	(i)	B will not be able to decrypt it // A's private key would be needed to decrypt it // only A could decrypt it; (as ...) Only A has access to A's private key // B cannot access A's private key; **MAX 1**	1
	(b)	(ii)	As A's public key is available to anyone; Anybody could decrypt it; **MAX 1**	1

5	(c)		**Subject-related points:** **Purpose:** To authenticate/confirm identity of sender // that message was sent by A // To detect if message has been tampered with/changed; **How used:** *[1]Hash/digest produced/calculated from message // (shortened) value calculated from message; ***A*** *message is hashed* ***A*** *message digest created* *[1]Hash encrypted with A's private key; *[1]Encrypted hash is known as the (digital) signature; *[2](Digital) signature is appended to message; ***A*** *transmitted with message* ***A*** *even if stated or implied that this is done after the encryption of the message using B's public key* ***A*** *hash or digest* A encrypts message and signature with B's public key; ***A*** *without reference to signature but* ***TO*** *if clear from order of statements or what candidate has written that the signature is not encrypted with B's public key* B decrypts message and signature with B's private key; ***A*** *without reference to signature* B decrypts (digital) signature using A's public key (to reveal hash); B reproduces/recalculates hash from received message; ***A*** *re-hashed* ***A*** *creates new digest* *[3]If received hash matches reproduced hash then message has not been tampered with // identity of sender is authenticated; **A** Data for message **A** Digest, checksum for hash **A** Encrypted hash/Encrypted digest for signature **A** Example of hashing method e.g. MD2/4/5/6, SH0/1/224/256/384/512 *[1] = as an alternative to these three points, allow one mark for the idea that the digital signature is calculated from/hashed from/a digest of the message *[2] = only award this mark if there is previously the concept of the hash or signature being produced. *[3] = can only be awarded if there is clear concept that the comparison is to a recalculated hash **Only one mark should be awarded for the purpose. Other marks must come from how the digital signature is used.** The purpose mark could be implicit in the how used mark and should be awarded if it is. It is acceptable for steps to be missed out. Accept responses with message sent from B to A if it is clear that this is what the candidate has done.	

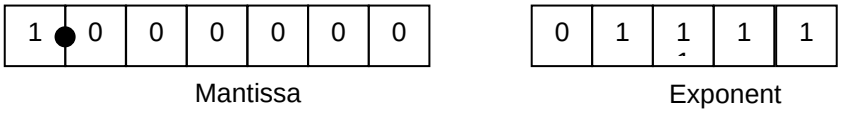
How to award marks:

Mark Bands and Description	
5-6	*To achieve a mark in this band, candidates must meet the subject criterion (SUB) and all 5 of the quality of language criteria (QWCx).* *SUB* Candidate has covered both the purpose and the use of digital signatures, and has made at least five subject-related points including both creation and use. **To get 6 marks, the answer must include reference to the encryption of the message digest/hash using A's private key.** *QWC1* Text is legible. *QWC2* There are few, if any, errors of spelling, punctuation and grammar. Meaning is clear. *QWC3* The candidate has selected and used a form and style of writing appropriate to the purpose and has expressed ideas clearly and fluently. *QWC4* Sentences (and paragraphs) follow on from one another clearly and coherently. *QWC5* Appropriate specialist vocabulary has been used.
3-4	*To achieve a mark in this band, candidates must meet the subject criterion (SUB) and 4 of the 5 quality of language criteria (QWCx).* *SUB* Candidate has provided a description of some parts of the process and has made at least three subject-related points. *QWC1* Text is legible. *QWC2* There may be occasional errors of spelling, punctuation and grammar. Meaning is clear. *QWC3* The candidate has, in the main, used a form and style of writing appropriate to the purpose, with occasional lapses. The candidate has expressed ideas clearly and reasonably fluently. *QWC4* The candidate has used well-linked sentences (and paragraphs). *QWC5* Appropriate specialist vocabulary has been used.
1-2	*To achieve a mark in this band, candidates must meet the subject criterion (SUB) and 4 of the 5 quality of language criteria (QWCx).* *SUB* Only one or two relevant points have been made. *QWC1* Most of the text is legible. *QWC2* There may be some errors of spelling, punctuation and grammar but it should still be possible to understand most of the response. *QWC3* The candidate has used a form and style of writing which has many deficiencies. Ideas are not always clearly expressed. *QWC4* Sentences (and paragraphs) may not always be well-connected. *QWC5* Specialist vocabulary has been used inappropriately or not at all.
0	Candidate has made no relevant points.

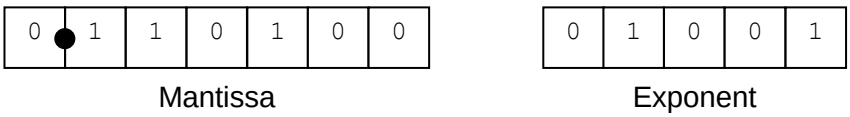
Note: Even if English is perfect, candidates can only get marks for the points made at the top of the mark scheme for this question.

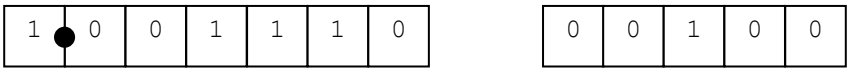
If a candidate meets the subject criterion in a band but does not meet the quality of language criteria then drop mark by one band, providing that at least 4 of the quality of language criteria are met in the lower band. If 4 criteria are not met then drop by two bands.

6

6	(a)		 **1 mark** for correct mantissa **1 mark** for correct exponent	2

6	(b)		**1 method mark** for either: • showing correct value of both mantissa and exponent in denary • showing binary point shifted 6 places to right in binary number • indicating that final answer calculated using answer = mantissa x $2^{exponent}$ Mantissa = -0.6875 // -11/16 Exponent = 6 Answer = -44 **1 mark** for correct answer *If answer is correct and some working has been shown, award two marks, even if working would not have gained credit on its own.* *Marks for working can be awarded in the answer.*	2

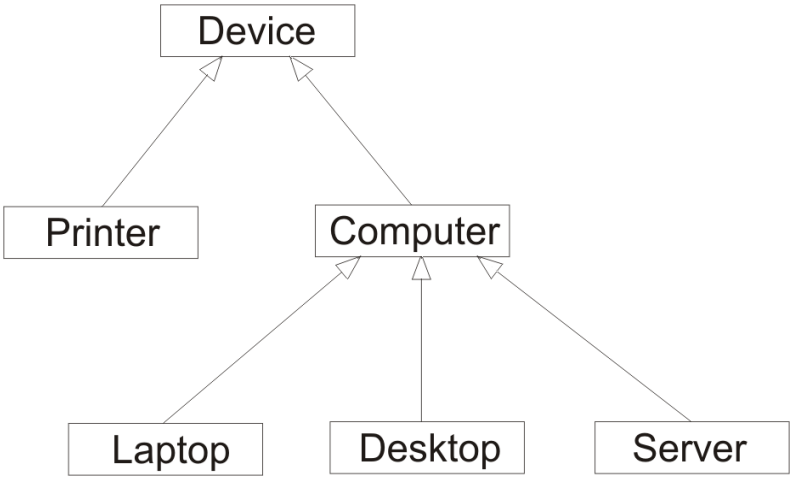
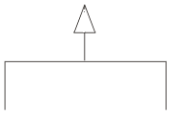
6	(c)		**1 mark** for working: Showing a bit pattern including 1101 and any number of preceding or following 0s, but no other 1s; Showing the correct value of the exponent in denary (9); Showing the binary point being shifted 9 places; **MAX 1** **1 mark** for correct mantissa and exponent together:  *If answer is correct and some working has been shown, award two marks, even if working would not have gained credit on its own.* *Marks for working can be awarded in the answer.*	2

6	(d)		**2 marks** for working: Correct representation of 12.5 in fixed point binary: 1100.1; Bits flipped: 0011.0 // 10011.0; **A** any number of preceding 1s Correct representation of -12.5 in fixed point twos complement: 10011.1; **A** any number of preceding 1s Showing the correct value of the exponent in denary (4) or binary // showing the binary point being shifted four places; Showing the correct value of the mantissa in floating point binary (1.001110) **MAX 2** **1 mark** for correct mantissa and exponent together:  Mantissa Exponent *If answer is correct and some working has been shown, award three marks, even if working would not have gained credit on its own.* *Marks for working can be awarded in the answer.*	3

6	(e)			

Calculation	Type of Error
Multiplying two very large numbers together.	Overflow;
Dividing a number by a very large number.	Underflow;
Adding together two numbers of very different sizes e.g. a tiny number to a very big number.	Cancellation;

If same answer is used more than once and it is correct in one instance then award the mark for the correct instance. 3

7	(a)		 **1 mark** for Device at top of diagram with Printer and Computer directly underneath it and linked to it and no other labels linked to it; **1 mark** for Computer with Laptop, Desktop and Server directly underneath it and linked to it, and no other labels linked to it (except Device above); **1 mark** for correctly styled diagram, i.e. lines drawn as arrows and boxes (any shape) around labels; - ***This mark is only available if candidate has already achieved at least one mark for correct contents of the diagram.*** **A** arrows drawn as:  **A** filled/empty arrowheads **A** diagram rotated by 90 degrees	3

7	(b)		Computer = Class/Subclass/Extends(Device) 1 (Public) Procedure AddDevice (Override) 1 Function GetProcessorName Function GetRAMCapacity 1 Function GetHDDCapacity Private / Protected ProcessorName : String RAMCapacity : Integer 1 HDDCapacity : Integer End Accept answers that use different notations, so long as meaning is clear. **1 mark** for correct header including name of class and parent class; **1 mark** for redefining the AddDevice procedure; **1 mark*** for defining all 3 extra functions needed to read variable values, all identified as being public (keyword public is optional if functions are declared before variables); **1 mark**# for defining all 3 extra variables, with appropriate data types and identified as being private; **A** any sensible numeric types for RAMCapacity and HDDCapacity, do not have to be whole numbers **A** answers that indicate separately that each variable is private or each method is public **R** do not award mark for declaring new functions if any of the functions have the same name as the variables **I** parameters to methods, minor changes to names that do not affect clarity * - Do not award this mark if any extra functions/procedures have been declared, except for functions that would set values e.g. SetProcessorName or an incorrectly named procedure to add e.g. AddComputer # - Do not award this mark if any extra variables have been declared	4

7	(c)		Laptop = Class/Subclass (Computer) 1 (Public) Procedure AddDevice (Override) 1 Function GetBluetoothInstalled Private / Protected 1 BluetoothInstalled : Boolean End **1 mark** for correct header including name of class and parent class; **MAX 1** of the following two marks: **1 mark** for redefining the AddDevice procedure; **1 mark*** for : • defining the GetBluetoothInstalled function needed to read this value, identified as being public (keyword public is optional if function is declared before variable) • defining the BluetoothInstalled variable with an appropriate data type as being private. **A** Boolean or whole number types for BluetoothInstalled but reject string, character or real number types **A** Different sensible name for GetBluetoothInstalled function e.g. CheckBluetoothInstalled, IsBluetoothInstalled **A** answers that indicate separately that each variable is private or each method is public **I** parameters to methods, minor changes to names that do not affect clarity **I** addition of any extra functions or variables * Do not award this mark if any extra functions / procedures / variables declared, except for a SetBluetoothInstalled procedure.	2

7	(d)		*What (2 marks):* Wireless/RF (protocol/standard/technology); For exchanging data over short distances // for creating Personal Area Network; **NE** "uses waves" for "wireless" *Example (1 mark):* Any sensible example, related to the use of Bluetooth <u>with the laptop</u> e.g. synchronising contacts between phone/laptop, sending photographs from phone to laptop, Bluetooth mouse, Bluetooth headset/headphones (used with laptop) etc; **NE** connecting to wireless network **NE** mouse If the example makes clear that the technology is wireless, but this is not explicitly stated in the "What" part of the response then the "Wireless" mark should be awarded in the "What" part.	3

8	(a)		System will be storing confidential/personal data (that must be kept securely/safely); Centralised/improved security management // centralised login system // centralised administration // administration will be easier; Centralised backup; Harder for users to change security/sharing settings; Running database from a server will avoid concurrency issues // will avoid problems if two users/computers update (a record in the) database simultaneously; **A** will allow simultaneous updates/access **R** answers that imply that on a peer-to-peer system there would be a separate copy of the database on each workstation Running database from server will ensure that it is always available (as server is unlikely to be turned off) // Files would always be available (as server is unlikely to be turned off); Server (operating system) may allow more simultaneous connections than a workstation // (operating system software on) workstations may not allow enough simultaneous connections for ten users; **NE** the database could be stored on the server **MAX 2**	2

8 (b)

Subject-related points:

How works:
All/most processing done by (central) server; ***A*** *all software run on server*
Keystrokes/mouse clicks/user input transmitted from workstation/terminal to server over network; ***A*** *workstations are just interfaces*
Image/data needed to produce image transmitted from server to terminal over network;
Applications not installed on (thin client) workstations // all applications on server;
Operating system loaded by clients from server at boot;
Selection of hardware:
Higher bandwidth network connection required;
Network must use switch not hub;
Slower processor /reduced RAM/ no HDD required in workstations; ***A*** *other examples of limited hardware requirements* ***A*** *'Dumb terminal'*
Server must have multiple processors/a lot of RAM;

NE more powerful / less powerful, higher performance / lower performance, cheaper / more expensive

Accept the opposite of points e.g. for "Slower processor" accept "a thick client system would need a faster processor".

How to award marks:

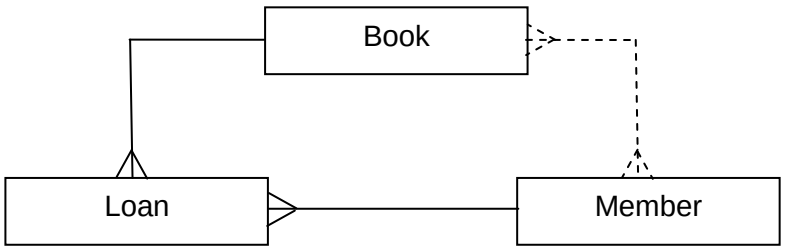
Mark Bands and Description	
4	*To achieve a mark in this band, candidates must meet the subject criterion (SUB) and all 5 of the quality of language criteria (QWCx).* *SUB* Candidate has covered both how a thin-client system works and how this affects the choice of hardware, and has made at least four subject-related points. *QWC1* Text is legible. *QWC2* There are few, if any, errors of spelling, punctuation and grammar. Meaning is clear. *QWC3* The candidate has selected and used a form and style of writing appropriate to the purpose and has expressed ideas clearly and fluently. *QWC4* Sentences (and paragraphs) follow on from one another clearly and coherently. *QWC5* Appropriate specialist vocabulary has been used.
3	*To achieve a mark in this band, candidates must meet the subject criterion (SUB) and 4 of the 5 quality of language criteria (QWCx).* *SUB* Candidate has covered both how a thin-client system works and how this affects the choice of hardware, and has made at least three subject-related points. *QWC1* Text is legible. *QWC2* There may be occasional errors of spelling, punctuation and grammar. Meaning is clear. *QWC3* The candidate has, in the main, used a form and style of writing appropriate to the purpose, with occasional lapses. The candidate has expressed

			ideas clearly and reasonably fluently. *QWC4* The candidate has used well-linked sentences (and paragraphs). *QWC5* Appropriate specialist vocabulary has been used.	
			1-2 *To achieve a mark in this band, candidates must meet the subject criterion (SUB) and 4 of the 5 quality of language criteria (QWCx).* *SUB* Candidate has covered one or both of how thin client systems work and how this affects hardware choice. Only one or two points are made. *QWC1* Most of the text is legible. *QWC2* There may be some errors of spelling, punctuation and grammar but it should still be possible to understand most of the response. *QWC3* The candidate has used a form and style of writing which has many deficiencies. Ideas are not always clearly expressed. *QWC4* Sentences (and paragraphs) may not always be well-connected. *QWC5* Specialist vocabulary has been used inappropriately or not at all.	
			0 Candidate has made no relevant points.	
			Note: Even if English is perfect, candidates can only get marks for the points made at the top of the mark scheme for this question. If a candidate meets the subject criterion in a band but does not meet the quality of language criteria then drop mark by one band, providing that at least 4 of the quality of language criteria are met in the lower band. If 4 criteria are not met then drop by two bands.	4

8	(c)		To connect networks using different protocols // to convert transmitted data from one protocol to another;	1

9	(a)		Composite (key); **A** Compound (key) **Note:** The word key is not required	1

9	(b)		Data is atomic // no repeating groups (of attributes); **R** No repeated columns/attributes/data/values No partial (key) dependencies // No (non-key) attribute depends on part of the primary key but not the whole of it // all non-prime attributes are (functionally) dependent on the whole of every candidate key // (non-key) attributes depend on the whole key; No non-key dependencies // No transitive dependencies // (non-key) attributes depend on nothing but the key; Every (non-key) attribute is dependent upon the key; Every determinant is a candidate key; **A** "field" for "attribute" **A** "part" for "partial" **MAX 2**	2

9	(c)		 **1 mark** per correct relationship (the dashed one is given) **MAX 1 if more than two relationships drawn**	2

9	(d)		**Solution 1:** SELECT EmailAddress, Forename, Surname FROM Book, Member, Loan WHERE Author = 'Lucas Bailey' AND Book.BookID=Loan.BookID AND Member.MemberID=Loan.MemberID **1 mark** for correct three fields in SELECT clause **1 mark** for correct three tables in FROM clause **1 mark** for Author = 'Lucas Bailey' **1 mark** for Book.BookID=Loan.BookID linked by AND **1 mark** for Member.MemberID=Loan.MemberID linked by AND **Solution 2:** SELECT EmailAddress, Forename, Surname FROM Book INNER JOIN Loan ON Book.BookID=Loan.BookID INNER JOIN Member on Member.MemberID=Loan.MemberID WHERE Author = 'Lucas Bailey' **1 mark** for correct three fields in SELECT clause **1 mark** for correct three tables in FROM clause **1 mark** for join from Member to Loan **1 mark** for join from Loan to Book **1 mark** for Author = 'Lucas Bailey' **Note:** Joins do not need to be done in same order as example Do not award mark for SELECT clause if extra attributes listed. Do not award mark for 'Lucas Bailey' unless it is enclosed in single or double quotation marks. Accept table names before fieldnames. Accept use of Alias/AS command e.g. FROM Member as M then use of M as table name. Accept insertion of spaces into fieldnames **DPT** for unnecessary punctuation – allow one semicolon at the very end of the statement, but not at the end of each clause. Also, allow insertion of brackets at logically allowable places in the WHERE/FROM clauses. **DPT** for fieldname before table name. **Refer responses using nested SQL queries to team leaders.**	5

9	(e)		*Alternative 1:* INSERT INTO Book VALUES (837023, “Kenyan Safari”, “Karen Matu”, “African Travel Guides”) *Alternative 2:* INSERT INTO Book (BookID, Title, Author, Publisher) VALUES (837023, “Kenyan Safari”, “Karen Matu”, “African Travel Guides”) **1 mark** for INSERT INTO Book; **1 mark** for correct field values. If alternative 2 is used, the order of the values and fieldnames must correspond to each other; The values Kenyan Safari, Karen Matu and African Travel Guides must be in single or double quotation marks for the mark to be awarded. Accept the value 837023 with or without quotation marks. **A** Minor errors in transcribing the data from the question into the answer. **A** omission of brackets	2

9	(f)		**ONE MARK FOR PRINCIPLE AND MAX TWO MARKS FOR IMPLEMENTATION** **Principle:** Create a new table (**A** link table) (BookCopy); through which Book and Loan tables will be (indirectly) linked; **Implementation details using a new primary key:** Create a new <u>unique ID/key field</u> (e.g. CopyID) (for each copy); Store the BookID and the CopyID in the new table; Replace the BookID in the Loans table with this CopyID; Note: In this implementation, CopyID is unique, i.e. BookID 1 and 2 cannot both have CopyID 1. **Implementation details using a composite key:** Create a new field CopyID; Composite key formed by BookID and CopyID; **TO** if composite key is clearly in book table or loan table Store the BookID and the CopyID in the new table; **R** adding CopyID to Book table as this would created data redundancy but this does not talk out other mark scheme points Add the CopyID field to Loans table; **R** replace BookID with CopyID Note: In this implementation, CopyID is not unique, e.g. BookID 1 and 2 can both have CopyID 1.	

			MAX 3 **Marks can be awarded for principle and/or implementation details.** **A** Relation for Table **A** Answers if candidates have rewritten new relations, awarding marks where the points above can be observed in the redrawn relations; **A** alternative name for CopyID	3

9	(g)	(i)	So that searching, adding and deleting can be done efficiently // To speed up searching, adding and deleting; **A** just one of searching, adding, deleting **NE** organise efficiently **NE** easily for efficiently	1
	(g)	(ii)	***Alternative 1 (context-specific):*** A function/calculation that computes a record position/address; within a specified range; from a key field value; **A** an example of a hashing function e.g. calculate an integer from certain letters in a field for one mark ***Alternative 2 (generic):*** A function (**A** algorithm) H, applied to a key k; which generates a hash value (H(k)) (of range smaller than the domain of values of k); **MAX 2**	2
	(g)	(iii)	***What is (1 mark):*** When more than one key value maps to the same record position/address // when two keys compute the same hash value; **A** "two records", "two items" or "two pieces of data" for "two keys" but **R** "two files" – both in this question part only ***How dealt with (1 mark):*** Store the record in the next available location in the file // store a pointer (in each file location) that points to a list of records that have all collided at the file location; **A** idea that each storage location could store more than one record e.g. five records per location, if explained. **A** example of what "next available" might be **A** key is rehashed **A** table for file	2

10	(a)		*Connected* // There is a path between each pair of vertices; *Undirected* // No direction is associated with each edge; *Has no cycles* // No (simple) circuits // No closed chains // No closed paths in which all the edges are different and all the intermediate vertices are different // No route from a vertex back to itself that doesn't use an edge more than once or visit an intermediate vertex more than once; **A** no loops **MAX 1** **Alternative definitions:** A simple cycle is formed if any edge is added to graph; Any two vertices can be connected by a unique simple path;	1

10	(b)		No route from entrance to exit / through maze; Maze contains a loop/circuit ; **A** more than one route through maze; Part of the maze is inaccessible / enclosed; **R** Responses that clearly relate to a graph rather than the maze **MAX 1**	1

10	(c)		<table><tr><th></th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>2</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>3</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>4</td><td>0</td><td>1</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>5</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td></tr><tr><td>6</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>7</td><td>0</td><td>0</td><td>0</td><td>0</td><td>1</td><td>0</td><td>0</td></tr></table> (allow some symbol in the central diagonal to indicate unused) **or** <table><tr><th></th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th></tr><tr><td>1</td><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>2</td><td></td><td>0</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>3</td><td></td><td></td><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr><tr><td>4</td><td></td><td></td><td></td><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><td>5</td><td></td><td></td><td></td><td></td><td>0</td><td>1</td><td>1</td></tr><tr><td>6</td><td></td><td></td><td></td><td></td><td></td><td>0</td><td>0</td></tr><tr><td>7</td><td></td><td></td><td></td><td></td><td></td><td></td><td>0</td></tr></table> (with the shaded portion in either half – some indication must be made that half of the matrix is not being used. This could just be leaving it blank, unless the candidate has also represented absence of an edge by leaving cells blank) **1 mark** for drawing a 7x7 matrix, labelled with indices on both axis and filled only with 0s and 1s, or some other symbol to indicate presence/absence of edge. e.g. T/F. Absence can be represented by an empty cell. **1 mark** for correct values entered into matrix, as shown above;	2

10	(d)	(i)	Routine defined in terms of itself // Routine that calls itself; **A** alternative names for routine e.g. procedure, algorithm **NE** repeats itself	1
10	(d)	(ii)	Stores return addresses; Stores parameters; Stores local variables; **NE** temporary variables Stores contents of registers; **A** To keep track of calls to subroutines/methods etc. **MAX 1** Procedures / invocations / calls must be returned to in reverse order (of being called); As it is a LIFO structure; **A** FILO As more than one / many return addresses / <u>sets of</u> values may need to be stored (at same time) // As the routine calls itself and for each call/invocation a new return address / new values must be stored; **MAX 1**	2

10	(e)			5

				Discovered							Completely Explored							
Call	**V**	**U**	**EndV**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**F**
	-	-	7	F	F	F	F	F	F	F	F	F	F	F	F	F	F	F
DFS(1,7)	1	2	7	**T**	F	F	F	F	F	F	F	F	F	F	F	F	F	F
DFS(2,7)	2	1	7	T	**T**	F	F	F	F	F	F	F	F	F	F	F	F	F
		3	7	T	T	F	F	F	F	F	F	F	F	F	F	F	F	F
DFS(3,7)	3	2	7	T	T	**T**	F	F	F	F	F	F	**T**	F	F	F	F	F
DFS(2,7)	2	4	7	T	T	T	F	F	F	F	F	F	T	F	F	F	F	F
DFS(4,7)	4	2	7	T	T	T	**T**	F	F	F	F	F	T	F	F	F	F	F
		5	7	T	T	T	T	F	F	F	F	F	T	F	F	F	F	F
DFS(5,7)	5	4	7	T	T	T	T	**T**	F	F	F	F	T	F	F	F	F	F
		6	7	T	T	T	T	T	F	F	F	F	T	F	F	F	F	F
DFS(6,7)	6	5	7	T	T	T	T	T	**T**	F	F	F	T	F	F	**T**	F	F
DFS(5,7)	5	7	7	T	T	T	T	T	T	F	F	F	T	F	F	T	F	F
DFS(7,7)	7	5	7	T	T	T	T	T	T	**T**	F	F	T	F	F	T	**T**	**T**
DFS(5,7)	5	-	7	T	T	T	T	T	T	T	F	F	T	F	**T**	T	T	T
DFS(4,7)	4	-	7	T	T	T	T	T	T	T	F	F	T	**T**	T	T	T	T
DFS(2,7)	2	-	7	T	T	T	T	T	T	T	F	**T**	T	T	T	T	T	T
DFS(1,7)	1	-	7	T	T	T	T	T	T	T	**T**	T	T	T	T	T	T	T

1 mark for having the correct values changes in each region highlighted by a rectangle and no incorrect changes in the region. Ignore the contents of any cells that are not changed.
A alternative indicators that clearly mean True and False.
A it is not necessary to repeat values that are already set (shown lighter in table)

11	(a)		Is it possible in general to ***write a program/algorithm;*** that can tell, given any program and its inputs and without ***running/executing the program;***, whether the given program with its given inputs will halt? **A** "it" in second reference to program. **A** "create a Turing machine" for "write an algorithm"	2

11	(b)		Shows that some problems are non-computable / undecideable // shows that some problems cannot be solved by a computer/algorithm; In general, inspection alone cannot always determine whether any given algorithm will halt for its given inputs // a program cannot be written that can determine whether any given algorithm will halt for its given inputs; **A** it is not computable **MAX 1**	1

12	(a)		ab^+c // abb^*c // ab^*bc; **I** ^ at start, $ at end of expression	1

12	(b)		$(0\|1)1^*$ // $(1\|0)1^*$ // [01]1*// [10]1* // [0|1]1* // [1|0]1* // 0 | $(0?1^+)$ **I** ^ at start, $ at end of expression	1

UMS conversion calculator www.aqa.org.uk/umsconversion