



Cambridge International AS & A Level

COMPUTER SCIENCE**9618/32**

Paper 3 Advanced Theory

May/June 2023

MARK SCHEME

Maximum Mark: 75

Published

This mark scheme is published as an aid to teachers and candidates, to indicate the requirements of the examination. It shows the basis on which Examiners were instructed to award marks. It does not indicate the details of the discussions that took place at an Examiners' meeting before marking began, which would have considered the acceptability of alternative answers.

Mark schemes should be read in conjunction with the question paper and the Principal Examiner Report for Teachers.

Cambridge International will not enter into discussions about these mark schemes.

Cambridge International is publishing the mark schemes for the May/June 2023 series for most Cambridge IGCSE, Cambridge International A and AS Level and Cambridge Pre-U components, and some Cambridge O Level components.

This document consists of **12** printed pages.

Generic Marking Principles

These general marking principles must be applied by all examiners when marking candidate answers. They should be applied alongside the specific content of the mark scheme or generic level descriptors for a question. Each question paper and mark scheme will also comply with these marking principles.

GENERIC MARKING PRINCIPLE 1:

Marks must be awarded in line with:

- the specific content of the mark scheme or the generic level descriptors for the question
- the specific skills defined in the mark scheme or in the generic level descriptors for the question
- the standard of response required by a candidate as exemplified by the standardisation scripts.

GENERIC MARKING PRINCIPLE 2:

Marks awarded are always **whole marks** (not half marks, or other fractions).

GENERIC MARKING PRINCIPLE 3:

Marks must be awarded **positively**:

- marks are awarded for correct/valid answers, as defined in the mark scheme. However, credit is given for valid answers which go beyond the scope of the syllabus and mark scheme, referring to your Team Leader as appropriate
- marks are awarded when candidates clearly demonstrate what they know and can do
- marks are not deducted for errors
- marks are not deducted for omissions
- answers should only be judged on the quality of spelling, punctuation and grammar when these features are specifically assessed by the question as indicated by the mark scheme. The meaning, however, should be unambiguous.

GENERIC MARKING PRINCIPLE 4:

Rules must be applied consistently, e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

GENERIC MARKING PRINCIPLE 5:

Marks should be awarded using the full range of marks defined in the mark scheme for the question (however; the use of the full mark range may be limited according to the quality of the candidate responses seen).

GENERIC MARKING PRINCIPLE 6:

Marks awarded are based solely on the requirements as defined in the mark scheme. Marks should not be awarded with grade thresholds or grade descriptors in mind.

Mechanics of Marking:

Every mark given should have a corresponding tick on the script. The number of ticks on each (part) question should match the number of marks awarded for that (part) question. If giving Benefit of Doubt, **the BOD must be accompanied by a tick.**

If a candidate has not given a response or the response is in no way related to the question, such as 'don't know', NR (the Hash key) should be awarded rather than zero.

Every part question must be annotated to show that it has been read even if awarding NR. Please ensure that all part questions that are marked as NR are also annotated with the SEEN icon. This is a requirement of RM3.

There are **two** blank pages at the start of each script and a page of syntax diagrams on page 4 (question 3) that must be annotated with the SEEN icon. 😞

NEW: Words or phrases that are underlined, must be present in the candidate's answer. Words or phrases that are emboldened indicate that the idea represented by the bold text must be included.

Even though the comments box is visible at the bottom of the screen, please do not put comments or question marks on the scripts. When scripts are returned to centres all the annotations including comments, are visible.

If work has been crossed out and something written in its place, the replacement work is marked even if the crossed-out work is correct. If the crossed-out work has not been replaced, mark the crossed-out answer. Please also annotate the unmarked work as SEEN, especially if the replacement answer is on a separate sheet.

For single mark answers, mark the first answer on the line, unless there is a note to the contrary on the mark scheme.

If a candidate writes something that is not enough (NE) for a mark, but is not actually incorrect, continue reading, even if the mark scheme says, for example, mark first two answers.

Question	Answer	Marks																																
1(a)	One mark per mark point • correct mantissa • correct exponent with associated working Answer <table><tr><th colspan="10">Mantissa</th><th colspan="6">Exponent</th></tr><tr><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td><td>1</td><td>1</td><td>0</td></tr></table> Working exponent = 6 (movement of 6 bicimal places seen to find what exponent should be) calculation of denary 6 to binary (000)110	Mantissa										Exponent						0	1	0	1	0	1	0	1	1	1	0	0	0	1	1	0	2
Mantissa										Exponent																								
0	1	0	1	0	1	0	1	1	1	0	0	0	1	1	0																			
1(b)	One mark per mark point (Max 3) MP1 the mantissa of the number would need to be 0.101011111001 / 13 bits / digits MP2 ... it can only store 10 bits / digits MP3 The 3 least significant digits would be truncated MP4 ...causing a loss of precision	3																																

Question	Answer	Marks								
2(a)	One mark per mark point (Max 3) MP1 records are stored in a particular order MP2 the order is determined based on the value in a key field MP3 records are accessed one after the other MP4 records can be found by searching from the beginning of the file, record by record, MP5 ... until the required record is found <u>or</u> key field value is exceeded.	3								
2(b)	One mark for each correct hash value (Max 2) <table><tr><th>Record key</th><th>Hash value</th></tr><tr><td>3003</td><td>3</td></tr><tr><td>1029</td><td>4</td></tr><tr><td>7630</td><td>0</td></tr></table>	Record key	Hash value	3003	3	1029	4	7630	0	2
Record key	Hash value									
3003	3									
1029	4									
7630	0									

Question	Answer	Marks
3(a)	One mark per correct valid/invalid and reason combination (Max 2) 9SW – Invalid Reason - This begins with a digit and a variable must begin with a letter UWY – Valid Reason – This begins with a letter and is followed by two other letters.	2
3(b)	One mark per mark point (Max 3) • <code><word> ::= <letter> </code> • <code>...<word><letter></code> • <code><variable> ::= <word> <word><digit></code> Example answers <code><word> ::= <letter> <word><letter></code> <code><word> ::= <letter><word> <letter></code> <code><variable> ::= <word> <word><digit></code> <code><variable> ::= <word><digit> <word></code>	3
3(c)(i)	Answer must be two letters followed by one, two or three digits using the letters and digits on the syntax diagram. Example answer AC768	1
3(c)(ii)	One mark per mark point (Max 3) • always has only two letters • one, two or three digits possible • correct arrows, boxes and name of syntax diagram Example answer Vehicle registration	3

Question	Answer	Marks												
4	One mark for each correct line connecting an OOP term to its description (Max 4). <table><thead><tr><th>OOP term</th><th>Description</th></tr></thead><tbody><tr><td>Encapsulation</td><td>methods used to return the value of a property</td></tr><tr><td>Getters</td><td>the process of putting data and methods together as a single unit</td></tr><tr><td>Polymorphism</td><td>methods used to update the value of a property</td></tr><tr><td>Setters</td><td>allows methods to be redefined for derived classes</td></tr><tr><td></td><td>enables the defining of a new class that inherits from a parent class</td></tr></tbody></table>	OOP term	Description	Encapsulation	methods used to return the value of a property	Getters	the process of putting data and methods together as a single unit	Polymorphism	methods used to update the value of a property	Setters	allows methods to be redefined for derived classes		enables the defining of a new class that inherits from a parent class	4
OOP term	Description													
Encapsulation	methods used to return the value of a property													
Getters	the process of putting data and methods together as a single unit													
Polymorphism	methods used to update the value of a property													
Setters	allows methods to be redefined for derived classes													
	enables the defining of a new class that inherits from a parent class													

Question	Answer	Marks
5(a)	One mark per mark point (Max 2) MP1 to produce a virtually unbreakable encryption system / send virtually un-hackable secure messages ... MP2 ...using the laws / principles of quantum mechanics / properties of photons MP3 detects eavesdropping ... MP4 ...because the properties of photons change MP5 to protect security of data transmitted over fibre optic cables MP6 to enable the use of longer keys.	2
5(b)	One mark per mark point (Max 3) MP1 Symmetric cryptography uses a single key to encrypt and decrypt messages, Asymmetric cryptography uses two. MP2 The symmetric key is shared, whereas with asymmetric, only the public key is shared (and the private key isn't). MP3 ... the risk of compromise is higher with symmetric encryption and asymmetric encryption is more secure. MP4 Symmetric cryptography is a simple process that can be carried out quickly, but asymmetric is much more complex, so slower. MP5 The length of the keys in symmetric encryption are (usually) shorter than those for asymmetric (128/256 bits v 2048 bits).	3

Question	Answer	Marks
6(a)	One mark for TYPE TAppointments and ENDTYPE correct One mark for every two correct declarations (Max 3) Example answer TYPE TAppointments DECLARE Name : STRING DECLARE DateOfBirth : DATE DECLARE Telephone : STRING DECLARE LastAppointment : DATE DECLARE NextAppointment : DATE DECLARE TreatmentsComplete : BOOLEAN ENDTYPE	4
6(b)	One mark for each correctly completed line (Max 5) <pre> DECLARE DentalRecord : ARRAY[1:250] OF TAppointments DECLARE DentalFile : STRING DECLARE Count : INTEGER DentalFile ← "DentalFile.dat" OUTPUT "The file ", DentalFile, " contains these records:" OPENFILE DentalFile FOR RANDOM Count ← 1 REPEAT SEEK DentalFile, Count GETRECORD DentalFile, DentalRecord[Count] OUTPUT DentalRecord[Count] Count ← Count + 1 UNTIL EOF(DentalFile) CLOSEFILE DentalFile </pre>	5

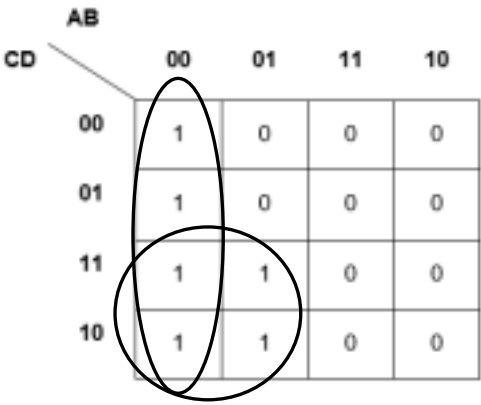
Question	Answer	Marks
7(a)	One mark per mark point (Max 2) MP1 Packet switching is most commonly used on data networks such as the internet to send large data files that don't need to be live streamed MP2 Packet switching is used when it is necessary to be able to overcome failed/faulty lines by rerouting. MP3 Packet switching is used when it is necessary for the communication to be more secure. MP4 Packet switching is used for high volume data transmission. MP5 Packet switching is used when it isn't necessary to use all the bandwidth. MP6 Specific examples e.g. email, text messages, documents, VOIP etc. (up to two marks).	2

Question	Answer	Marks
7(b)	One mark per mark point (Max 4) MP1 Circuit switching uses a dedicated channel to make communication, whereas packet switching forms data into packets to transmit over a digital network. MP2 The dedicated path for circuit switching must be established before the transfer of data can commence, which is not the case with packet switching (as it doesn't require a dedicated path). MP3 Data in packet switching is split into packets, in circuit switching the message remains intact. MP4 All of the transmission in circuit switching follows the same path whereas different packets in packet switching can take different routes. MP5 The message is received in the same order in which it is sent with circuit switching, but with packet switching, the packets can be received out of order (for assembly at the destination). MP6 Circuit switching is implemented at the physical layer while packet switching is implemented at the network layer. MP7 Circuit switching uses the whole bandwidth of the channel used, packet switching can share bandwidth. MP8 Circuit switching communication ends with an error but packet switching allows packets to be re-sent. MP9 Circuit switching is a simpler process than packet switching.	4

Question	Answer	Marks
8(a)	One mark per mark point (Max 2) MP1 Pipelining allows several instructions to be processed simultaneously / concurrently. MP2 ... therefore, increasing the CPU instruction throughput / the number of instructions completed per unit of time. MP3 Each instruction stage / subtask is completed during one clock cycle MP4 No two instructions can execute their same stage of instruction / subtask at the same clock cycle. MP5 ... e.g., while one instruction is being decoded, the next instruction can be fetched, etc.	2

Question	Answer	Marks																																																																																																																																																																																												
8(b)	One mark per mark point (Max 4) • First stage of first instruction in first clock cycle • First instruction successfully completed over five clock cycles • Remaining instructions completed over ten clock cycles ... • ... with no extra instruction sections added on any row. Example answer 1 <table><tr><td colspan="2"></td><th colspan="12">Clock cycles</th></tr><tr><td colspan="2"></td><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th></tr><tr><td rowspan="5">Processor stages</td><th>IF</th><td>1.1</td><td>2.1</td><td>3.1</td><td>4.1</td><td>5.1</td><td>6.1</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th>ID</th><td></td><td>1.2</td><td>2.2</td><td>3.2</td><td>4.2</td><td>5.2</td><td>6.2</td><td></td><td></td><td></td><td></td><td></td></tr><tr><th>OF</th><td></td><td></td><td>1.3</td><td>2.3</td><td>3.3</td><td>4.3</td><td>5.3</td><td>6.3</td><td></td><td></td><td></td><td></td></tr><tr><th>IE</th><td></td><td></td><td></td><td>1.4</td><td>2.4</td><td>3.4</td><td>4.4</td><td>5.4</td><td>6.4</td><td></td><td></td><td></td></tr><tr><th>WB</th><td></td><td></td><td></td><td></td><td>1.5</td><td>2.5</td><td>3.5</td><td>4.5</td><td>5.5</td><td>6.5</td><td></td><td></td></tr></table> Example answer 2 <table><tr><td colspan="2"></td><th colspan="12">Clock cycles</th></tr><tr><td colspan="2"></td><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th>9</th><th>10</th><th>11</th><th>12</th></tr><tr><td rowspan="5">Processor stages</td><th>IF</th><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td></td><td></td><td></td><td></td><td></td><td></td></tr><tr><th>ID</th><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td></td><td></td><td></td><td></td><td></td></tr><tr><th>OF</th><td></td><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td></td><td></td><td></td><td></td></tr><tr><th>IE</th><td></td><td></td><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td></td><td></td><td></td></tr><tr><th>WB</th><td></td><td></td><td></td><td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td></td><td></td></tr></table>			Clock cycles														1	2	3	4	5	6	7	8	9	10	11	12	Processor stages	IF	1.1	2.1	3.1	4.1	5.1	6.1							ID		1.2	2.2	3.2	4.2	5.2	6.2						OF			1.3	2.3	3.3	4.3	5.3	6.3					IE				1.4	2.4	3.4	4.4	5.4	6.4				WB					1.5	2.5	3.5	4.5	5.5	6.5					Clock cycles														1	2	3	4	5	6	7	8	9	10	11	12	Processor stages	IF	A	B	C	D	E	F							ID		A	B	C	D	E	F						OF			A	B	C	D	E	F					IE				A	B	C	D	E	F				WB					A	B	C	D	E	F			4
		Clock cycles																																																																																																																																																																																												
		1	2	3	4	5	6	7	8	9	10	11	12																																																																																																																																																																																	
Processor stages	IF	1.1	2.1	3.1	4.1	5.1	6.1																																																																																																																																																																																							
	ID		1.2	2.2	3.2	4.2	5.2	6.2																																																																																																																																																																																						
	OF			1.3	2.3	3.3	4.3	5.3	6.3																																																																																																																																																																																					
	IE				1.4	2.4	3.4	4.4	5.4	6.4																																																																																																																																																																																				
	WB					1.5	2.5	3.5	4.5	5.5	6.5																																																																																																																																																																																			
		Clock cycles																																																																																																																																																																																												
		1	2	3	4	5	6	7	8	9	10	11	12																																																																																																																																																																																	
Processor stages	IF	A	B	C	D	E	F																																																																																																																																																																																							
	ID		A	B	C	D	E	F																																																																																																																																																																																						
	OF			A	B	C	D	E	F																																																																																																																																																																																					
	IE				A	B	C	D	E	F																																																																																																																																																																																				
	WB					A	B	C	D	E	F																																																																																																																																																																																			

Question	Answer	Marks																									
9(a)	One mark for every two correct products (Max 3) (Z =) $\bar{A}\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}C\bar{D} + \bar{A}\bar{B}CD + \bar{A}BC\bar{D} + \bar{A}BCD$	3																									
9(b)	Two marks if no errors present One mark if one error present AB CD <table><tr><td>00</td><td>00</td><td>01</td><td>11</td><td>10</td></tr><tr><td>00</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>01</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><td>11</td><td>1</td><td>1</td><td>0</td><td>0</td></tr><tr><td>10</td><td>1</td><td>1</td><td>0</td><td>0</td></tr></table>	00	00	01	11	10	00	1	0	0	0	01	1	0	0	0	11	1	1	0	0	10	1	1	0	0	2
00	00	01	11	10																							
00	1	0	0	0																							
01	1	0	0	0																							
11	1	1	0	0																							
10	1	1	0	0																							

Question	Answer	Marks
9(c)	One mark for each correct loop (Max 2) 	2
9(d)	One mark for each mark point (Max 2) - Any correct Boolean term - Boolean terms and operator correct and no other terms present $(Z =) \bar{A}\bar{B} + \bar{A}C$	2
9(e)	One mark for simplest form (Max 1) $(Z =) \bar{A} (\bar{B} + C)$	1

Question	Answer	Marks
10(a)	One mark from: Supervised (learning) Unsupervised (learning) Reinforcement (learning) Deep (learning)	1

Question	Answer	Marks																																																									
10(b)	One mark for each correct calculation as follows (Max 2): - Node B (from Begin) (Line 3 in table) - Node C (from Begin) (Line 4 in table) - Node D (from Begin) (Line 5 in table) One mark for each correct calculation as follows (Max 2): - Node G (from C) (Line 6 in table) - Node F and Node End (from G) (Lines 7 and 8 in table) - Node End (from F) (Line 9 in table) One mark for correct path (Max 1): - Begin $\Rightarrow$ C $\Rightarrow$ G $\Rightarrow$ F $\Rightarrow$ End <table><tr><th>Start node</th><th>Destination node</th><th>Cost from start node (g)</th><th>Heuristic (h)</th><th>Total (f = g + h)</th></tr><tr><td>Begin</td><td>Begin</td><td>0</td><td>12</td><td>12</td></tr><tr><td>Begin</td><td>A</td><td>5</td><td>8</td><td>13</td></tr><tr><td>Begin</td><td>B</td><td>6</td><td>7</td><td>13</td></tr><tr><td>Begin</td><td>C</td><td>5</td><td>7</td><td>12</td></tr><tr><td>Begin</td><td>D</td><td>4</td><td>11</td><td>15</td></tr><tr><td>C</td><td>G</td><td>5 + 2 = 7</td><td>5</td><td>12</td></tr><tr><td>G</td><td>F</td><td>5 + 2 + 4 = 11</td><td>1</td><td>12</td></tr><tr><td>G</td><td>End</td><td>5 + 2 + 7 = 14</td><td>0</td><td>14</td></tr><tr><td>F</td><td>End</td><td>5 + 2 + 4 + 1 = 12</td><td>0</td><td>12</td></tr><tr><td></td><td></td><td></td><td></td><td></td></tr></table> <table><tr><td>Final Path</td><td>Begin $\Rightarrow$ C $\Rightarrow$ G $\Rightarrow$ F $\Rightarrow$ End</td></tr></table>	Start node	Destination node	Cost from start node (g)	Heuristic (h)	Total (f = g + h)	Begin	Begin	0	12	12	Begin	A	5	8	13	Begin	B	6	7	13	Begin	C	5	7	12	Begin	D	4	11	15	C	G	5 + 2 = 7	5	12	G	F	5 + 2 + 4 = 11	1	12	G	End	5 + 2 + 7 = 14	0	14	F	End	5 + 2 + 4 + 1 = 12	0	12						Final Path	Begin $\Rightarrow$ C $\Rightarrow$ G $\Rightarrow$ F $\Rightarrow$ End	5
Start node	Destination node	Cost from start node (g)	Heuristic (h)	Total (f = g + h)																																																							
Begin	Begin	0	12	12																																																							
Begin	A	5	8	13																																																							
Begin	B	6	7	13																																																							
Begin	C	5	7	12																																																							
Begin	D	4	11	15																																																							
C	G	5 + 2 = 7	5	12																																																							
G	F	5 + 2 + 4 = 11	1	12																																																							
G	End	5 + 2 + 7 = 14	0	14																																																							
F	End	5 + 2 + 4 + 1 = 12	0	12																																																							
Final Path	Begin $\Rightarrow$ C $\Rightarrow$ G $\Rightarrow$ F $\Rightarrow$ End																																																										

Question	Answer	Marks															
11(a)(i)	One mark for every two correct identifiers (Max 2) <table border="1"> <thead> <tr> <th>Identifier</th><th>Data type</th><th>Description</th></tr> </thead> <tbody> <tr> <td>Queue</td><td>STRING</td><td>An array to store the contents of the queue.</td></tr> <tr> <td>RearPointer</td><td>INTEGER</td><td>Points to the last term of the queue.</td></tr> <tr> <td>Length</td><td>INTEGER</td><td>Indicates the number of items in the queue.</td></tr> <tr> <td>FrontPointer</td><td>INTEGER</td><td>Points to the first term of the queue.</td></tr> </tbody> </table>	Identifier	Data type	Description	Queue	STRING	An array to store the contents of the queue.	RearPointer	INTEGER	Points to the last term of the queue.	Length	INTEGER	Indicates the number of items in the queue.	FrontPointer	INTEGER	Points to the first term of the queue.	2
Identifier	Data type	Description															
Queue	STRING	An array to store the contents of the queue.															
RearPointer	INTEGER	Points to the last term of the queue.															
Length	INTEGER	Indicates the number of items in the queue.															
FrontPointer	INTEGER	Points to the first term of the queue.															
11(a)(ii)	One mark for each correctly completed line (Max 5) <pre> CONSTANT MaxLength = 50 DECLARE FrontPointer : INTEGER DECLARE RearPointer : INTEGER DECLARE Length : INTEGER DECLARE Queue : ARRAY[0:MaxLength - 1] OF STRING // Initialisation of queue PROCEDURE Initialise FrontPointer ← -1 RearPointer ← -1 Length ← 0 ENDPROCEDURE // Adding a new item to the queue PROCEDURE Enqueue(NewItem : STRING) IF Length < MaxLength THEN // IF Length <= MaxLength - 1 THEN RearPointer ← RearPointer + 1 IF RearPointer > MaxLength - 1 THEN RearPointer ← 0 ENDIF Queue[RearPointer] ← NewItem Length ← Length + 1 ENDIF ENDPROCEDURE </pre>	5															
11(b)	One mark per mark point (Max 3) • Print jobs are expected to be actioned by the printer in the order they are received • ... because the printer queue is a queue, the first job to be sent to the printer would be the first job printed. • If the printer queue was on a stack, the first job the printer received would not be printed until all the other jobs have been printed.	3															