CAMBRIDGE
International Education

Cambridge International AS & A Level

COMPUTER SCIENCE **9618/33**

Paper 3 Advanced Theory **May/June 2025**

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 2025 series for most Cambridge IGCSE, Cambridge International A and AS Level components, and some Cambridge O Level components.

This document consists of **16** 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 descriptions 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.

Annotations guidance for centres

Examiners use a system of annotations as a shorthand for communicating their marking decisions to one another. Examiners are trained during the standardisation process on how and when to use annotations. The purpose of annotations is to inform the standardisation and monitoring processes and guide the supervising examiners when they are checking the work of examiners within their team. The meaning of annotations and how they are used is specific to each component and is understood by all examiners who mark the component.

We publish annotations in our mark schemes to help centres understand the annotations they may see on copies of scripts. Note that there may not be a direct correlation between the number of annotations on a script and the mark awarded. Similarly, the use of an annotation may not be an indication of the quality of the response.

The annotations listed below were available to examiners marking this component in this series.

Annotations

Annotation	**Meaning**
✓	Correct
✗	Incorrect
λ	To indicate where a key word/phrase/code is missing.
≀	Not relevant or used to separate parts of an answer.
〰	Indicates a part of the answer that is incorrect.
Highlighter	To draw attention to a particular aspect or to indicate where parts of an answer have been combined.
TV	Too vague.
REP	Repetition
NE	No examples or not enough.
BOD	Benefit of Doubt.

Annotation	**Meaning**
NAQ	Not Answered Question.
SEEN	Indicates that work or a page has been seen including blank answer spaces and blank pages.
FT	Follow through.
I	Ignore

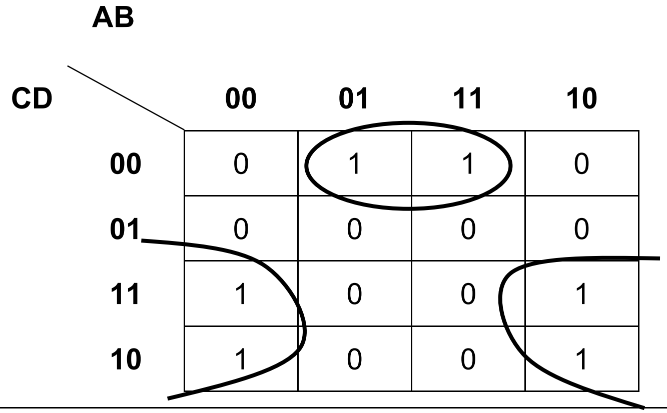
Question	Answer	Marks
1(a)	**One** mark per mark point MP1 `TYPE VideoLibrary` and `ENDTYPE` correct MP2 Declare used correctly for every field in the response MP3 All Three STRING fields correct (shaded) MP4 Remaining Three fields correct (unshaded) Example answer	**4**

```
TYPE VideoLibrary
    DECLARE VideoID : STRING
    DECLARE Title : STRING
    DECLARE ReleaseYear : INTEGER
    DECLARE PurchaseDate : DATE
    DECLARE VideoFormat : STRING
    DECLARE RunningTime : INTEGER
ENDTYPE
```

Question	Answer	Marks
1(b)	**One** mark for identification of field and **one** mark for reason `VideoFormat` This field can have a fixed range of possible values	**2**

Question	Answer	Marks
2(a)	**One** mark for two's complement version and **one** mark for denary version **Mantissa** 0 1 1 1 1 1 1 1 **Exponent** 0 1 1 1 1 1 1 1 127×2^{120} // 0.9921875×2^{127} // $127/128 \times 2^{127}$	**2**
2(b)	**One** mark per mark point for working (**Max 2**) • number converted to binary e.g., positive binary version of 3.59375 = (0)11.10011 • negative two's complement version - bits flipped and 1 added = 100.01101 • -4 + 1/4 + 1/8 + 1/32 // -4 + 025 +0.125 + 0.03125 // -(64 + 32 + 16 + 2 + 1)/32 • $-2^{2} + 2^{-2} + 2^{-3} + 2^{-5}$ • $2^{2} (- 2^{0} + 2^{-4} + 2^{-5} + 2^{-7})$ **One** mark per mark point • correct mantissa • correct exponent, with working seen. **Mantissa** 1 0 0 0 1 1 0 1 **Exponent** 0 0 0 0 0 0 1 0	**4**

Question	Answer	Marks
3(a)	**Two** marks for all **six** correct terms **One** mark for any **four** correct terms $\mathbf{Z} = \overline{A}.\overline{B}.C.\overline{D} + \overline{A}.\overline{B}.C.D + \overline{A}.B.\overline{C}.\overline{D} + A.\overline{B}.C.\overline{D} + A.\overline{B}.C.D + A.B.\overline{C}.\overline{D}$	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>3(b)(i)</td><td>Two marks if no errors present
One mark if one error present
<table>
<tr><th>CD \ AB</th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><td>00</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>01</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>11</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>10</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
</table></td><td>2</td></tr>
<tr><td>3(b)(ii)</td><td>One mark for each correct loop (Max 2)

<table>
<tr><th>CD \ AB</th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><td>00</td><td>0</td><td>1</td><td>1</td><td>0</td></tr>
<tr><td>01</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>11</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
<tr><td>10</td><td>1</td><td>0</td><td>0</td><td>1</td></tr>
</table></td><td>2</td></tr>
</table>

Question	Answer	Marks
3(b)(iii)	**One** mark for each mark point - One correct Boolean term - Boolean terms and operator correct and no other terms present **Z =** $\overline{B}.C + B.\overline{C}.\overline{D}$ // $B.\overline{C}.\overline{D} + \overline{B}.C$	**2**

Question	Answer	Marks
4(a)	**One** mark per mark point (**Max 5**) **One** mark per mark point for **purpose** of Internet Layer (**Max 3**) MP1 To identify the intended network and host. MP2 To add header containing IP addresses. // To address packets with their source and destination IP Addresses. MP3 To prepare packets for delivery by formatting them into datagrams. MP4 To route datagrams through the optimum route over a network. **One** mark per mark point for **purpose** of Link Layer (**Max 3**) MP5 To prepare the next hop by managing the link between two directly connected devices. // To identify and move traffic across local segments. MP6 To format datagrams into frames for transmission. MP7 To identify network protocols in the packet header. // To ensure correct network protocols are/is followed. MP8 To deliver frames to the receiving network. // To receive frames from the sending network. // To map IP addresses to MAC physical addresses. MP9 To provide error checking/error correction/recovery/reconstruction MP10 including resend requests.	**5**
4(b)	**One** mark for each correct marking point **(Max 4)** MP1 The router reads the IP address of the destination from the packet header. MP2 A router uses a routing table to find information … MP3 … about e.g., available hops / netmask / gateway used / adjacent routers / the status of the routers along the route. MP4 The router determines the next hop / optimum route. // The router sends the packet on its next hop. MP5 The router manages the hop counter // The hop counter is reduced by 1 every time the packet passes a router.	**4**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5</td><td>One mark per correct answer (Max 4)
<table>
<tr><th>OOP Term</th><th>Purpose</th></tr>
<tr><td>Getter</td><td>A method that accesses the value of a property</td></tr>
<tr><td>Setter</td><td>A method that changes the value of a property</td></tr>
<tr><td>Object</td><td>An instantiation / instance of a class</td></tr>
<tr><td>Method</td><td>A programmed function / procedure / subroutine / subprogram defined as part of a class</td></tr>
</table>
</td><td>4</td></tr>
</table>

Question	Answer	Marks
6(a)	**One** mark for each mark point (**Max 2**) • Process scheduling is required to ensure that all processes are executed in a timely manner • … and enables multitasking/multiprogramming/multiprocessing • … to minimise CPU idle time • … to ensure that no process is starved of resources. // … to ensure fair access to resources. • … ensures jobs/processes are completed in order of priority.	**2**

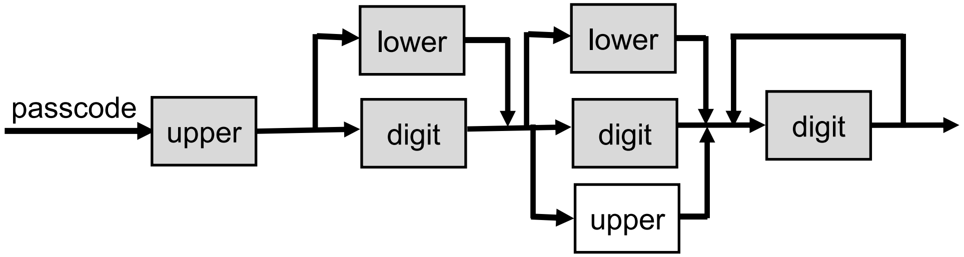
Question	Answer	Marks
6(b)	**One** mark for each mark point (**Max 3**) MP1 Processes are queued as they arrive. MP2 It is a pre-emptive scheduling routine. MP3 A fixed time quantum is given to each process. // Each process has an equal time slice. MP4 When a time slice ends, the status of the process is saved/queued so it can continue from where it left off in its next time slice. MP5 and the next process is executed for its time slice; its previous state is reinstated/restored, if applicable. MP6 If a process completes within its time slice, the next process is executed for its time slice. **One** mark for benefit (**Max 1**) MP7 Reduces average response time by limiting each process to a fixed amount of time. MP8 No issue with starvation of resources.	**4**

Question	Answer	Marks
7	**One** mark for each correct marking point (**Max 4**) MP1 An SSL/TLS connection is initiated by an application/client. MP2 Every new session begins with a handshake as defined by the SSL/TLS protocols. MP3 The client requests the digital certificate from the server // The server sends the digital certificate to the client. MP4 The client verifies the server's digital certificate MP5 … and obtains the server's public key. MP6 The encryption algorithms are agreed. // The symmetric session keys are generated/defined. MP7 A secure session is established between client and server.	**4**

Question	Answer	Marks
8	**One** mark per mark point (**Max 4**) MP1 It is the second stage of compilation // It's the compilation stage after lexical analysis. MP2 It takes input from the lexical analyser in the form of token streams. MP3 The source code is analysed / parsed against the rules of the language to detect any errors in the code. MP4 The output from this phase is a parse tree. MP5 Syntax errors are reported.	**4**

Question	Answer	Marks
9(a)	The second character must come from either lower or digit. It can't be upper.	**1**

9(b)	**Max 3**
	One mark
	MP1 `<upper> ::= J | K | L | V | X | Z`
	Either: fully written out answer
	MP2 Any two correct options for `<passcode>`
	MP3 Remaining two options correct for `<passcode>`
	Example answer
	`<passcode> ::=`
	`<upper><lower><lower><digit>|`
	`<upper><lower><digit><digit>|`
	`<upper><digit><digit><digit>|`
	`<upper><digit><lower><digit>`
	Or: answer with interim expression
	MP4 `<passcode> ::= <upper><middle><middle><digit>`
	MP5 `<middle> ::= <lower>|<digit>`
		3

9(c)	**One** mark per mark point MP1 Box for upper in correct place with correct connections MP2 Correct repetition arrow for final digit  passcode upper lower digit lower digit upper digit	**2**

Question	Answer	Marks
10(a)	To find the path between two points on a graph using the algorithm.	**1**
10(b)	**One** mark per point (**Max 2**) MP1 A* tries to find a better path (between two points) by using a heuristic function // A* finds the adjacent route with the shortest path and continues this until the destination is reached MP2 … Dijkstra's just explores all possible routes. MP3 The heuristic function on the A* algorithm gives priority to nodes that are supposed to be better than others / less costly than others. MP4 Dijkstra's algorithm cannot work with negative values/weights //A* algorithm can work with negative values/weights.	**2**
10(c)	**One** mark for each mark point (**Max 3**) MP1 Unsupervised learning uses algorithms to analyse / cluster MP2 … unlabelled data sets MP3 They discover hidden patterns / data groupings / clusters without the need for human intervention. MP4 It is able to discover similarities and differences in data / information.	**3**

Question	Answer	Marks
11	**One** mark for each correctly completed line	**5**

```
DECLARE Location : INTEGER
DECLARE NewStock : STRING
DECLARE CurrentStock : STRING
DECLARE Stored : BOOLEAN
DECLARE Max : INTEGER
Max ← 100000
Stored ← FALSE
Location ← 1
OPENFILE "StockList.dat" FOR RANDOM
OUTPUT "Enter the new item you wish to store"
INPUT NewStock
WHILE NOT Stored AND Location <= Max
   SEEK "StockList.dat", Location
   GETRECORD "StockList.dat", CurrentStock
   IF CurrentStock = "" THEN
      PUTRECORD "StockList.dat", NewStock
      Stored ← TRUE
   ELSE
      Location ← Location + 1
   ENDIF
ENDWHILE
IF Stored = FALSE THEN
   OUTPUT "The new stock item has not been stored as the file was full"
ENDIF
CLOSEFILE "StockList.dat"
```

Question	Answer	Marks
12(a)	**One** mark for any **two** correct ADTs (**Max 1**) • Binary tree • Graph • Linked list • Queue • Stack	**1**
12(b)	**One** mark for each marking point (**Max 4**) MP1 Temporary assignment of the element being 'inserted' before inner loop MP2 Appropriate inner loop MP3 Check if current `DataArray` content is > `Value` MP4 Moving data to adjacent element as required MP5 Appropriate updating of `Position` variable MP6 Re-insertion of the element outside the inner loop **Example algorithm**	**4**

```
FOR Index ← 2 to 1000
   Value ← DataArray[Index]
   Position ← Index – 1
   IF DataArray[Position] > Value THEN
      WHILE Position >= 1 AND DataArray[Position] > Value
         DataArray[Position + 1] ← DataArray[Position]
         Position ← Position – 1
      ENDWHILE
      DataArray[Position + 1] ← Value
   ENDIF
NEXT Index
```

Question	Answer	Marks
12(c)	**One** mark for each marking point MP1 The performance of a sorting routine should improve if the data is already partially sorted // The performance of a sorting routine is likely to be worse if the data is completely out of order MP2 The sort may take longer if the number of items to be sorted is larger // The sort may take less time if the number of items to be sorted is fewer.	**2**

Question	Answer	Marks
13	**One** mark for each marking point MP1 Correct `Index` and `Target` columns MP2 Correct `Numbers[5]` column MP3 Correct `Numbers[6]` and `Numbers[7]` columns MP4 Correct `Numbers[8]` column and no incorrect data added to any other of columns `[1, 2, 3, 4, 9, 10]`	**4**

		Numbers									
Index	**Target**	**[1]**	**[2]**	**[3]**	**[4]**	**[5]**	**[6]**	**[7]**	**[8]**	**[9]**	**[10]**
1	15	2	3	7	11	15	17	19	23	0	0
2											
3											
4											
5						17					
6							19				
7								23			
8									0		
9											