CAMBRIDGE
International Education

Cambridge International AS & A Level

COMPUTER SCIENCE **9618/32**

Paper 3 Advanced Theory **October/November 2024**

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 October/November 2024 series for most Cambridge IGCSE, Cambridge International A and AS Level components, and some Cambridge O Level components.

This document consists of **15** printed pages.

PUBLISHED

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.

Question	Answer	Marks
1(a)	**One** mark per mark point (**Max 3**) MP1 The data to be transmitted is divided into equal sized packets MP2 A packet header is attached to each packet containing key information MP3 … such as source/destination IP addresses, packet number, etc MP4 Packets are transmitted independently MP5 … and may travel though different routes/paths to the destination MP6 Routes are determined using a routing table//Packets take the optimum route depending on congestion MP7 The packets usually arrive out of order MP8 The packets are reassembled in the correct order at the destination // The packets are re-ordered using the sequence number/the header MP9 If packets are missing/corrupted a re-transmission request is sent / packets are re-sent.	**3**
1(b)	**One** mark for each benefit (**Max 2**) MP1 Packets are more likely to arrive because they can be re-routed if a problem occurs with one of the routes//Packets are more likely to arrive because if a packet is lost, it can be re-transmitted MP2 Bandwidth can be shared allowing packets from different messages to share the same path MP3 Considered secure as the packets generally travel via different routes MP4 High data transmission rate is possible **One** mark for each drawback (**Max 2**) MP5 Time delay because packets need to be re-ordered/reassembled at the destination//Time delay caused by missing packets needing to be re-sent//Time delay because it has to share the bandwidth of the circuit / channel with other packets MP6 Requires a complex algorithm to function MP7 Needs lots of RAM to handle large amounts of data.	**4**

Question	Answer	Marks
2(a)	**One** mark per mark point (**Max 2**) MP1 Records are stored one after the other as they are collected // records are stored in chronological order MP2 New records are appended to the end of the file.	2
2(b)	**One** mark for a use (**Max 1**) MP1 Creating unsorted / temporary transaction files MP2 Creating data logging files	**1**

Question	Answer	Marks
3(a)	**One** mark per mark point (**Max 3**) MP1 A user-defined record data type is a composite data type MP2 It uses other data types in its definition to form a single new data type MP3 The data types referenced may be primitive data types from a programming language or they may be other user-defined data types. MP4 Includes related items MP5 Includes a fixed number of items.	**3**

3(b)	**One** mark for `TYPE Order` and `ENDTYPE` correct
	One mark for correct use of `DECLARE` in all declarations seen
	One mark for the **two** shaded declarations
	One mark for the **two** unshaded declarations
	
	Example answer
	
	`TYPE Order`
	`  DECLARE AccountNumber : INTEGER`
	`  DECLARE OrderNumber : INTEGER`
	`  DECLARE OrderPrice : REAL`
	`  DECLARE OrderDate : DATE`
	`ENDTYPE`
		4

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>4(a)</td><td>One mark for working

• application of exponent to mantissa to go from 0.10001110111 to 01000111.0111 //
moving the binary point 7 places //
multiplying by $2^7/128$ in the fractions method //
64 + 4 + 2 +1 + .25+ .125 + .0625 seen

One mark for correct answer

• 71.4375
// $71^7/_{16}$</td><td>2</td></tr>
<tr><td>4(b)</td><td>One mark per mark point (Max 2)

• correct mantissa – exact answer only
• correct exponent – exact answer only

Mantissa
<table><tr><td>1</td><td>0</td><td>0</td><td>1</td><td>1</td><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td><td>1</td><td>0</td></tr></table>
Exponent
<table><tr><td>0</td><td>1</td><td>1</td><td>0</td></tr></table>
One mark per mark point for working (Max 2)

• number converted to binary e.g., positive binary version of 49.1875 = 0110001.0011 //
two’s complement version bits flipped and 1 added = 1001110.1101 //
–64 + 8 + 4 + 2 + .5 + .25 + .0625 //
–64 + 14.8125
• use of the exponent e.g. moving the binary point 6 places / × 2^6.</td><td>4</td></tr>
</table>

Question	Answer	Marks
5(a)	**One** mark for each correct name (**Max 2**) **One** mark for each correct corresponding expansion (**Max 2**) MP1 HTTP/HTTPS MP2 For sending and receiving / transferring web pages / hypertext MP3 FTP MP4 For sending and receiving files over a network / between devices MP5 POP3 MP6 Pull protocol / for receiving / downloading emails MP7 IMAP MP8 Pull protocol / for receiving / downloading emails MP9 SMTP MP10 Push protocol / for sending / uploading emails MP11 BitTorrent MP12 Peer-to-peer file sharing over a network	**4**
5(b)	**One** mark per mark point (**Max 3**) MP1 The application layer provides access to all the programs that exchange data // Interacts directly with user. MP2 … used by, for example, web browsers, server software. MP3 Communicates/enables data transfer to/from Transport layer // It allows applications to access the services used in other TCP/IP layers. MP4 It defines the protocols that any application uses to allow the exchange of data.	**3**

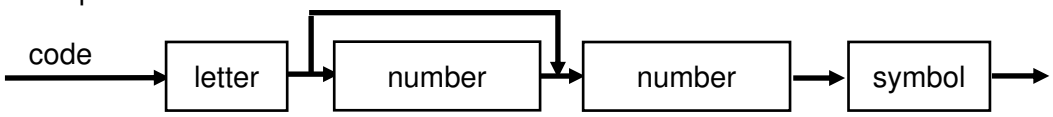
<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>6(a)</td><td>Three marks for all eight correct products and no additional products
Two marks for five, six or seven correct products
One mark for three or four correct products

$(X =) \overline{A}.\overline{B}.\overline{C}.D + \overline{A}.\overline{B}.C.\overline{D} + \overline{A}.B.\overline{C}.D. + \overline{A}.B.C.\overline{D} + A.\overline{B}.\overline{C}.D + A.\overline{B}.C.\overline{D} + A.B.\overline{C}.D + A.B.C.\overline{D}$</td><td>3</td></tr>
<tr><td>6(b)</td><td>Two marks if no errors present
One mark if one error present

<table>
<tr><th></th><th>AB</th><th></th><th></th><th></th></tr>
<tr><th>CD</th><th>00</th><th>01</th><th>11</th><th>10</th></tr>
<tr><td>00</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>01</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
<tr><td>11</td><td>0</td><td>0</td><td>0</td><td>0</td></tr>
<tr><td>10</td><td>1</td><td>1</td><td>1</td><td>1</td></tr>
</table></td><td>2</td></tr>
<tr><td>6(c)</td><td>One mark for each correct loop (Max 2)

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

Question	Answer	Marks
6(d)	**One** mark for each mark point (**Max 2**) MP1 Any correct relevant Boolean term MP2 Boolean terms with correct operator + and no other terms present $(X =)\ C.\overline{D} + \overline{C}.D\ //\ \overline{C}.D. + C.\overline{D}$	**2**

Question	Answer	Marks
7(a)	**One** mark per mark point (**Max 2)** MP1 21 - a number must begin with an odd digit, 2 is even MP2 123 - a number can only be one or two digits in length not three	**2**
7(b)	**One** mark per mark point (**Max 2)** MP1 `<symbol> ::= % | £ | # | @ | $` MP2 `<number> ::= <odd>|<odd><even>|<odd><odd>`	**2**
7(c)(i)	**One** mark per mark point (**Max 3)** MP1 letter, number and symbol all included in correct order: letter first, followed by number, finishing with symbol MP2 provision for one or two numbers including relevant connectors MP3 all other connections and label correct and no additional data Example answer 	**3**

Question	Answer	Marks

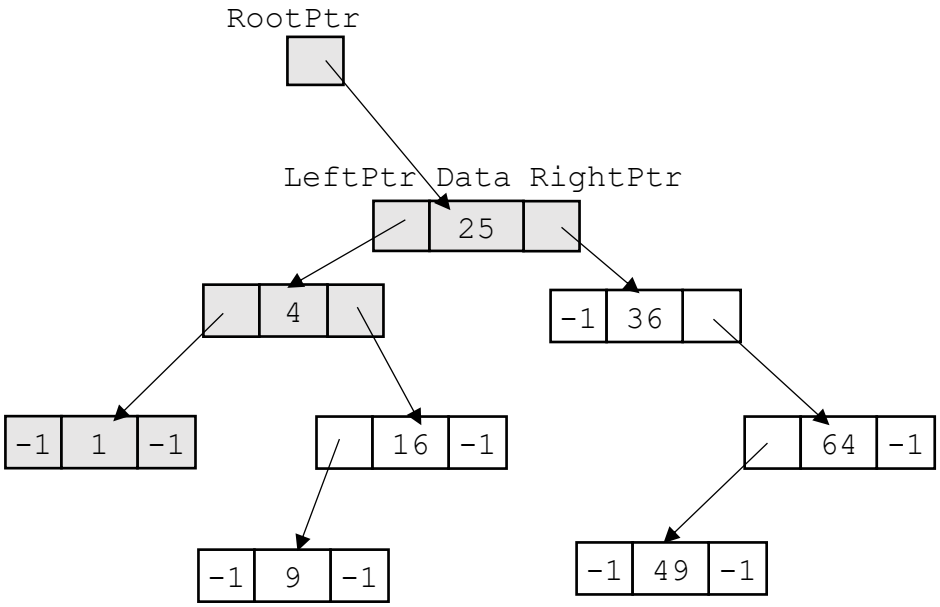
7(c)(ii)	**One** mark per correct line (**Max 2)**
	
	`<code> ::= <letter><number><symbol>|`
	`<letter><number><number><symbol>`
	
	OR
	
	`<code> ::= <letter><number><number><symbol>|`
	`<letter><number><symbol>`
	
	Alternative Answer
	
	One mark per correct line (**Max 2)**
	
	`<digits> ::= <number>|<number><number>`
	`<code> ::= <letter><digits><symbol>`
		2

Question	Answer	Marks
8	**One** mark per mark point (**Max 4)** MP1 many instruction formats possible MP2 large instruction set MP3 many addressing modes available MP4 uses variable length/multi-operation instructions MP5 multi-clock cycle instructions MP6 complex decoding of instructions MP7 uses complex circuits MP8 makes frequent use of cache memory MP9 uses programmable control unit // uses micro-programmed control unit // uses hardwired control unit MP10 hardware needs to be able to handle more complex instructions convert into sub-instructions // Design emphasis is on the hardware.	**4**

Question	Answer	Marks
9(a)	**One** mark per mark point (**Max 2)** MP1 the kernel receives a signal when an interrupt is generated MP2 the kernel checks the priority and reviews the status/priority of the current interrupts MP3 system enters kernel mode if the type of interrupt is of higher priority than the current process MP4 the kernel consults the interrupt dispatch table / IDT MP5 … and saves the state of the interrupted process / contents of the registers on the kernel stack MP6 the kernel restores the process state e.g. contents of registers once the interrupt is serviced	**2**
9(b)(i)	**One** mark per mark point (**Max 1)** MP1 multi-tasking allows computers to carry out / seem to carry out more than one process at a time	**1**
9(b)(ii)	**One** mark per mark point (**Max 2)** MP1 processor time/common hardware and resources is/are shared between tasks MP2 scheduling is used to decide on the processes to be carried out to ensure multi-tasking operates correctly / efficiently / without clashes MP3 one task of a higher priority can interrupt another task that is currently running	**2**

Question	Answer	Marks
10(a)	**One** mark per mark point (**Max 3)** MP1 Attributes/properties … MP2 … with their data types // … are variables bound to the class MP3 Methods … MP4 … that are subroutines / functions / procedures that act upon the attributes MP5 Getters/setters … MP6 … are methods that can fetch / update the contents of attributes MP7 A constructor … MP8 … that is used to create instances / objects of this class.	**3**

Question	Answer	Marks
10(b)	**One** mark per mark point (**Max 3)** MP1 A class is only defined once but many objects can be created from that class // A class is a template / blueprint from which objects are created // An object is an instance of a class MP2 No memory is allocated when a class is defined, but objects are allocated memory space whenever they are created MP3 A class cannot be manipulated as it is not available in the memory, but objects can be manipulated MP4 A class is defined but an object is declared / created / instantiated. MP5 A class can use inheritance. An object cannot.	**3**

Question	Answer	Marks
11(a)	**One** mark per mark point (**Max 4)** MP1 Five additional nodes with correct data values MP2 Correct null pointers in all nodes (7) – no extra null pointers where the arrow points to the next node MP3 Correct arrows to represent pointers joining parent nodes to child nodes – must come from the correct left or right pointer not the middle MP4 All nodes in correct order and no additional data in left/right pointer boxes. 	**4**

Question	Answer	Marks
11(b)	First **eight** rows of table (**Max 3**) **Three** marks for all eight correct rows **Two** marks for six or seven correct rows **One** mark for three, four or five correct rows Last row of table and FreePtr (**Max 1**) **One** mark for correct `FreePtr` with last row of table completely blank	**4**

RootPtr: 0

Index	LeftPtr	Data	RightPtr
0	1	25	2
1	3	4	4
2	-1	36	5
3	-1	1	-1
4	6	16	-1
5	7	64	-1
6	-1	9	-1
7	-1	49	-1
8			

FreePtr: 8

Question	Answer	Marks

11(c)	**One** mark for any correct row (**Max 4**)
	`FUNCTION SearchList(Item : INTEGER) RETURNS INTEGER`
	`  NullPtr ← -1`
	`  NowPtr ← RootPtr`
	`  WHILE NowPtr <> NullPtr`
	`  IF LinkList[NowPtr].Data < Item THEN`
	`  NowPtr ← LinkList[NowPtr].RightPtr`
	`  ELSE`
	`  IF LinkList[NowPtr].Data > Item THEN`
	`  NowPtr ← LinkList[NowPtr].LeftPtr`
	`  ELSE`
	`  RETURN NowPtr`
	`  ENDIF`
	`  ENDIF`
	`  ENDWHILE`
	`  RETURN NullPtr`
	`ENDFUNCTION`
		4