Cambridge Assessment International Education

Cambridge International AS & A Level

COMPUTER SCIENCE **9618/12**

Paper 1 Theory Fundamentals **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.

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:

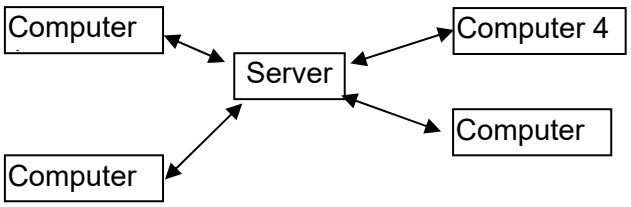
Rules must be applied consistently, e.g. in situations where candidates have not followed instructions or in the application of generic level descriptors.

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.

Question	Answer	Marks
1(a)	**1 mark** each to **max 2** • Allows the sharing of files/data // Allows communication between the devices • Allows the sharing of resources e.g. hardware / software (applications) • Allows central management // by example, backup, security, etc.	**2**
1(b)	**1 mark** each to **max 2** • Covers a small geographical area • The infrastructure is privately owned // not controlled by external organisations	**2**
1(c)	**1 mark** each to **max 2** • Each computer directly connected only to the server ... • ... <u>all</u> components correctly labelled  	**2**
1(d)	**1 mark** each to **max 3** • A protocol (suite) • For **data transmission** over standard / universal **wired** / **cabled** network connections • Uses Carrier Sense Multiple Access / Collision Detection (CSMA/CD) • Data is transmitted in frames • … each frame has a source and destination (IP/MAC) address • ... and error checking data (so damaged frames can be resent)	**3**
1(e)	**1 mark** each • The server performs **minimal** / **some processing** for the client • The clients also do **most** of their own processing/work independently // most of the resources are installed locally	**2**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>2(a)</td><td>1 mark for each correct feature or description

<table>
<tr><th>Feature</th><th>Description</th></tr>
<tr><td>Data dictionary</td><td>Data about the data in the database // data about the structure of the database // metadata for a database</td></tr>
<tr><td>Query processor</td><td>Software that allows the user to enter criteria, then finds and returns the appropriate result // software that processes and executes queries written in SQL</td></tr>
<tr><td><u>Logical schema</u></td><td>A model of a database that is not specific to one DBMS</td></tr>
<tr><td><u>Developer interface</u></td><td>A software tool that allows the user to create items such as tables, forms and reports</td></tr>
</table></td><td>4</td></tr>
<tr><td>2(b)</td><td>1 mark each to max 3

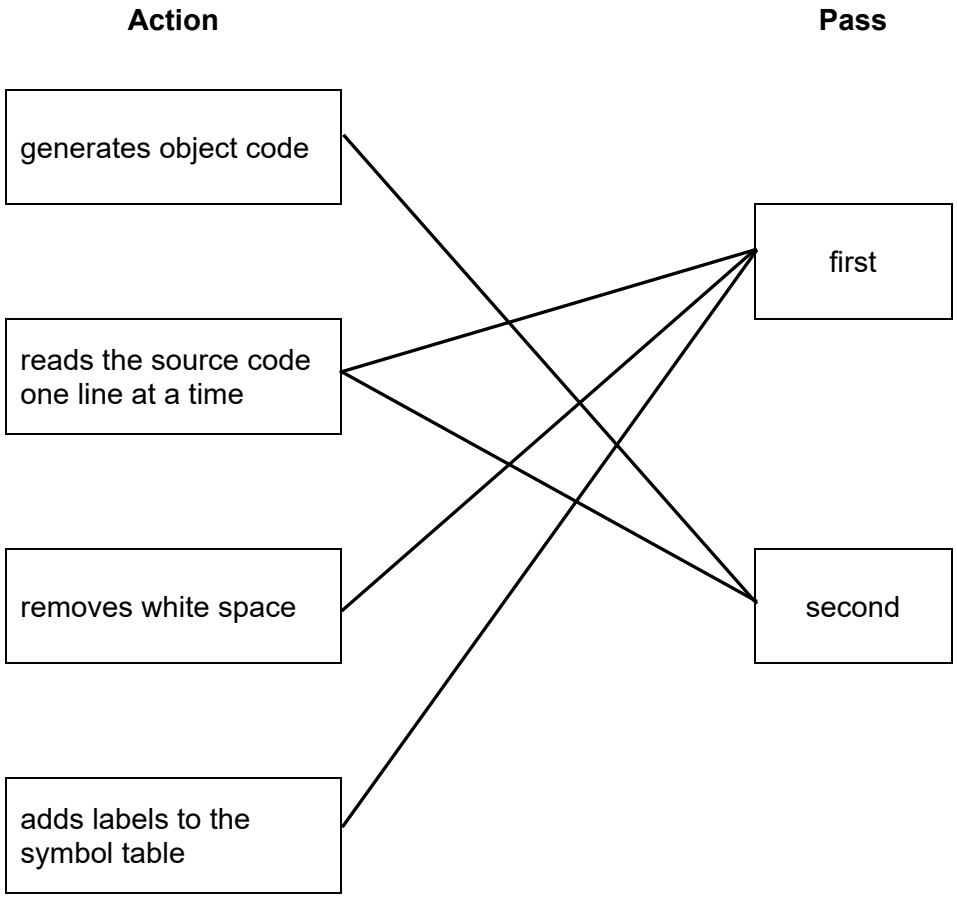
• Referential Integrity makes sure data is consistent
• Referential Integrity makes sure all data is up-to-date
• Referential integrity ensures that every foreign key has a corresponding primary key
• Referential Integrity prevents records from being added / deleted / modified incorrectly
• Referential Integrity makes sure that if data is changed in one place the change is reflected in all related records
• Referential Integrity makes sure any queries return accurate and complete results</td><td>3</td></tr>
<tr><td>2(c)(i)</td><td>1 mark each to max 2

• Presence check to make sure that the (rider level) is entered
• Look-up / Existence check to make sure the rider level is only Beginner, Intermediate or Advanced
• Length check to make sure the rider level entered is either 8 or 12 characters
• Type check to make sure the rider level is alphanumeric</td><td>2</td></tr>
</table>

Question	Answer	Marks

2(c)(ii)	**1 mark** each
	
	• SELECT field Name
	• FROM table HORSE
	• WHERE with Intermediate / Beginner
	• OR with Beginner / Intermediate
	
	Example answer:
	`SELECT Name`
	`FROM HORSE`
	`WHERE HorseLevel = "Intermediate"`
	`OR HorseLevel = "Beginner";`
		4

2(c)(iii)	**1 mark** each • SUM should be COUNT // SELECT COUNT(STUDENT.RiderLevel) • The WHERE statement needs the table names before each field name // WHERE STUDENT.StudentID = LESSON.StudentID • The OR should be AND // AND Date = #09/09/2023# • Beginner is missing the speech marks // STUDENT.RiderLevel = "Beginner";	**4**

Question	Answer	Marks
3(a)	**1 mark** for generates object code to second pass **1 mark** for reads source code one line at a time to both boxes **1 mark** for removes white space and adds labels to first pass 	**3**
3(b)	**1 mark** for each correct term **Direct** addressing is when the operand holds the memory address of the data. **Indirect** addressing is when the operand holds a memory address that stores the memory address of the data. **Immediate** addressing is when the operand is the data.	**3**

Question	Answer	Marks
4(a)	2^{16} // 65536	**1**
4(b)	**1 mark** for working; **1 mark** for answer • Working: +120 = 0111 1000 • Answer: 1000 0111	**2**

Question	Answer	Marks

4(c)	**1 mark** for working; **1 mark** for answer
	
	• Working: A04 = (10 * 16^2) + 4
	// A04 = (10 * 256) + 4
	// A04 = 1010 0000 0100
	
	• Answer: 2564
		2

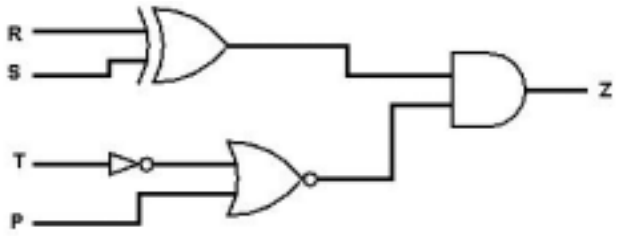
4(d)	0011 1100	**1**

Question	Answer	Marks
5(a)	**1 mark** each to **max 2** Examples: • Interrupt • Timing • Read • Write	**2**
5(b)	**1 mark** for description; **1 mark** for corresponding explanation Examples • Increase number of cores • Each core can **independently** carry out a process at the same time // so that more instructions are performed **in parallel** • Increase RAM capacity • ... allowing more applications to reside in memory at the same time, saving disk access times • Increase cache memory • More data can be stored in fast access so less time is spent accessing from RAM • Increase clock speed • More Fetch-Decode-Execute (FDE) cycles can run each second / **per unit time**	**4**
5(c)(i)	**1 mark** for a correct answer • 1 bit is transferred at a time • Can be synchronous **or** asynchronous • USB-3 is full duplex **and** earlier versions are half-duplex	**1**
5(c)(ii)	**1 mark** for identification of a suitable port Examples • HDMI • DisplayPort	**1**

<table>
<tr><th>Question</th><th>Answer</th><th>Marks</th></tr>
<tr><td>5(d)(i)</td><td>1 mark each to max 5

• Manages the scheduling of processes // decides which order to run processes
• Manages which resources the processes require
• … such as allocating memory
• Enables processes to share data
• Prevents interference between processes // resolution of conflicts
• Handles the process queue
• It allows multi-tasking / multi-processing
• ... by ensuring fair access, handling priorities and handling interrupts</td><td>5</td></tr>
<tr><td>5(d)(ii)</td><td>1 mark each to max 2

• To help users to set-up / configure / analyse / optimise / maintain the computer ...
• ... by for example, making memory allocation more efficient
• ... by for example, checking the system for faults</td><td>2</td></tr>
<tr><td>6(a)</td><td>1 mark for correct XOR and AND gates, with correct inputs
1 mark for correct NOT and NOR gates with correct inputs

 
</td><td>2</td></tr>
<tr><td>6(b)</td><td>1 mark for first 4 rows
1 mark for last 4 rows

<table>
<tr><th>P</th><th>Q</th><th>R</th><th>Working space</th><th>Z</th></tr>
<tr><td>0</td><td>0</td><td>0</td><td></td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>1</td><td></td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>0</td><td></td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td><td></td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td></td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td><td></td><td>0</td></tr>
<tr><td>1</td><td>1</td><td>0</td><td></td><td>1</td></tr>
<tr><td>1</td><td>1</td><td>1</td><td></td><td>1</td></tr>
</table></td><td>2</td></tr>
</table>

Question	Answer	Marks
7(a)(i)	**1 mark** each to max 2 • Set of pre-written / pre-compiled / pre-tested subroutines • ... which can be **called** in other programs • ... by **installing/importing** the library	**2**
7(a)(ii)	**1 mark** for each bullet point. Mark in pairs; **1 mark** for a benefit and **1 mark** for an appropriate expansion • (main) memory requirements for program is reduced • ... as dynamic link library is loaded only once / when required • the executable file size of the program using the DLL will be smaller • ... because the executable does not contain (all) the library routines • maintenance not needed to be done by the programmer • ... because the DLL is separate from program • no need to **recompile** the main program when changes are made to DLL • ... because changes / improvements/ error correction to the DLL file code are done independently of the main program • A single DLL file can be made available to several application programs • ... Saving space in memory / easing the pressure on memory	**4**
7(b)	No mark for choice. **1 mark** each to **max 3** for justification Interpreter • Allows the developer to make real-time changes • … so the program can be debugged **at each stage** • ... the effect of any changes made by the developer can be seen immediately • The developer can test when incomplete • … so small parts can be tested without having to test the rest of the program • … if one section does not work others can still be tested • To avoid dependent errors Compiler • The developer can debug multiple errors simultaneously • Produces an executable file • … so that the developer can test the program multiple times without recompiling	**3**

Question	Answer	Marks
7(c)	**1 mark** each Breakpoints: • **Stop the code** at a specific line to check the current progress / values Dynamic syntax checks: • Highlight / underline / colour syntax errors **as the code is entered** Context-sensitive prompts: • Suggest the code to add // automatically complete statements Single stepping: • Run the code **one line at a time** so the values can be checked	**4**
7(d)	**1 mark** each to **max 3** • Uses speech recognition • ... which identifies key phrases / words spoken • … and matches these to a database • ... and generates the most likely sentence / command / word	**3**