Paper 1 mark scheme

Question number	Answer	Additional guidance	Mark
1(a)	**A** bit ***B*** *is not correct because a bit is smaller than a byte* ***C*** *is not correct because a bit is smaller than a kibibyte* ***D*** *is not correct because a bit is smaller than a nibble*		**(1)**

Question number	Answer	Additional guidance	Mark
1(b)	**D** 32 ***A*** *is not correct because 5 bits can represent 32 values* ***B*** *is not correct because 5 bits can represent 32 values* ***C*** *is not correct because 5 bits can represent 32 values*		**(1)**

Question number	Answer	Additional guidance	Mark
1(c)(i)	**C** Unsigned integers store more positive values ***A*** *is not correct because unsigned integers are not more accurate* ***B*** *is not correct because overflow errors can still occur with unsigned integers* ***D*** *is not correct because the use of a parity bit is not relevant to the scenario*		**(1)**

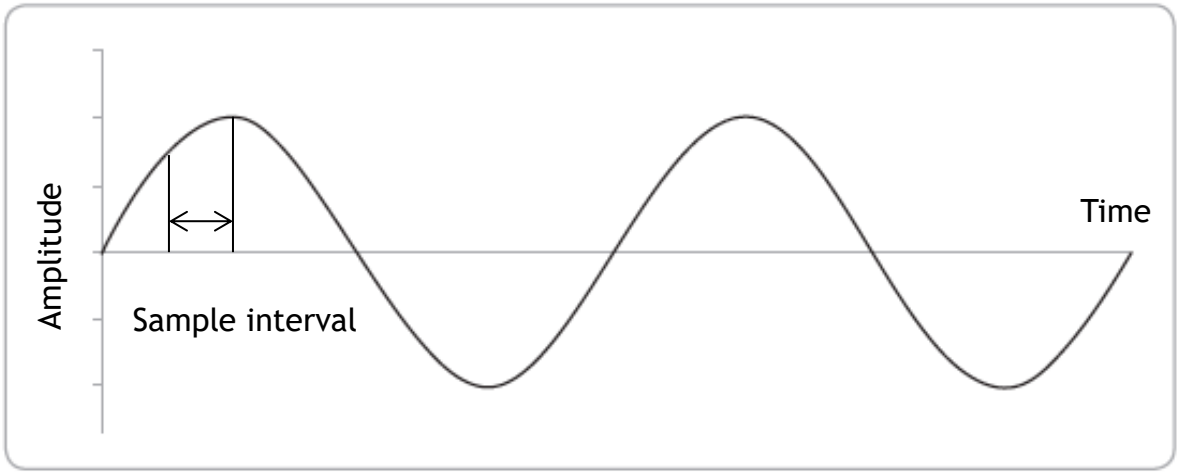
Question number	Answer	Additional guidance	Mark
1(c)(ii)	6 x 1024^4 Award **1** mark for sight of 1024 Award **1** mark for sight of ^{4} (applied only to 1024) Award **1** mark for sight of x6 Award **all** marks if the result of the calculation is given: 6,597,069,766,656 (bytes). **N.B.** This is not needed or expected.	Equivalent expressions are awarded.	**(3)**

Question number	Answer	Additional guidance	Mark
1(d)	**A** 1001+1000 ***B*** *is not correct because it will result in 1100* ***C*** *is not correct because it will result in 1110* ***D*** *is not correct because it will result in 1111*		**(1)**

Question number	Answer	Additional guidance	Mark
1(e)	0101 0010	Award 1 mark for each nibble in the correct location.	**(2)**

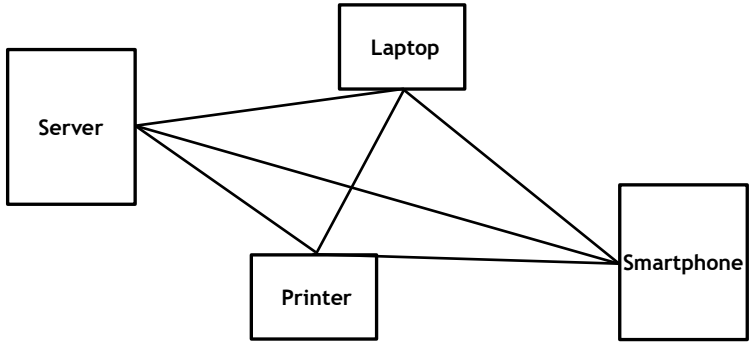
Question number	Answer	Additional guidance	Mark
1(f)(i)	3D	Award 1 mark for each nibble in the correct location.	**(2)**

Question number	Answer	Additional guidance	Mark
1(f)(ii)	Award **1** mark for the identification of the reason (1) with a linked justification/exemplification (1), up to a maximum of 2 marks. Hexadecimal is used as shorthand for binary / uses fewer digits/characters (1), so **humans** make fewer mistakes / find it easier to read/understand/remember/manipulate (1).	Do not accept answers suggesting that fewer digits save storage/memory.	**(2)**

Question number	Answer	Additional guidance	Mark
1(g)	X axis correctly labelled (1) Y axis correctly labelled (1) Sample interval indicated (1) Do not award marks if wavelength is labelled rather than sample interval. Accept for sample interval if two points given that are shorter than the wavelength.  	Accept a unit of time for the X axis. Accept signal strength for the Y axis.	**(3)**

Question number	Answer	Additional guidance	Mark
1(h)	Award **1** mark for: Sight of: 64 × 48 x 12 Award **1** mark for: Sight of: 1024 x 8 Award **1** mark for: Sight of: (1024) × (1024) **OR** $(1024)^2$ Award **1** mark for correct numerator/denominator orientation. Examples of expressions that gains full marks: $\frac{64 \times 48 \times 12}{1024 \times 1024 \times 8}$	Accept any other equivalent mathematical expression.	**(4)**

Question number	Answer	Additional guidance	Mark
2(a)	WAN/Wide Area Network		**(1)**

Question number	Answer	Additional guidance	Mark
2(b)	 • **1** mark for at least two connections to each device. • **2** marks for three connections to each device.		**(2)**

Question number	Answer	Additional guidance	Mark
2(c)	Any **two** items from: • Destination address (1) • Source address (1) • Error checking field / check sum (1) • Time stamp (1) • Sequence number (1). Accept any other appropriate response.		**(2)**

Question number	Answer	Additional guidance	Mark
2(d)	A linked description that makes reference to the following points: Each packet has a sequence number (added at the sending end) (1) The packets are put back into (sequence) order (at the destination) (1).		**(2)**

Question number	Answer	Additional guidance	Mark
2(e)	Total number of bits to transfer: **1** mark for 20×1024^2 **1** mark for $\times 8$ Speed in bits per second: **1** mark for 2×1000000 Numerator/denominator: **1** mark for $$\frac{bits\ to\ transfer}{bits\ per\ second}$$ e.g.: $$\frac{20 \times 1024 \times 1024 \times 8}{2 \times 1000000}$$	Any equivalent expression to be awarded.	**(4)**

Question number	Answer	Additional guidance	Mark
3(a)	Any **two** issues from: • Waste materials end in landfill (1) • Dangerous toxins are released into the ground/water (1) • Waste is transported overseas (carbon emissions) (1).		**(2)**

Question number	Answer	Additional guidance	Mark
3(b)(i)	Ransomware		**(1)**

Question number	Answer	Additional guidance	Mark
3(b)(ii)	Award **1** mark for the identification of a way (1) with a linked justification/exemplification (1), up to a maximum of 2 marks. • Software may contain security bugs (1) because it is unpatched (1). • Anti-malware may not identify an attack (1) because the virus definitions are out of date (1). Accept any other appropriate response.		**(2)**

Question number	Answer	Additional guidance	Mark
3(c)	Award 1 mark for the identification of an ethical concern (1) with a linked justification/exemplification (1), up to a maximum of 2 marks. • The data may no longer be private (1) because companies may share it (1). • People may not realise their data is analysed (1) because it is unclear who owns the data (1). • People are willing to sacrifice (some) privacy (1) in return for access to services (1). Accept any other appropriate response.		**(2)**

Question number	Answer	Additional guidance	Mark
3(d)	Award 1 mark for the identification of a way (1) with a linked justification/exemplification (1), up to a maximum of 2 marks. • Students are deterred from unsafe practices (1) because consequences are clarified (1). • People follow safe/good practices (1) because permitted activities are defined/set out (1). Accept any other appropriate response.		**(2)**

Question number	Answer	Additional guidance	Mark
4(a)	Any **two** functions from: • Repairing files (1) • Compression (1) • Defragmentation (1) • Back-up (1) • Firewall (1) • Managing application updates (1) • Formatting disks/drives (1) • System analysis tools (1).		**(2)**

Question number	Answer	Additional guidance	Mark
4(b)	Any **one** way from: • By identifying bad programming practices (1) • By identifying vulnerabilities in the code (1) • By checking for efficiency of code (1).	Do not accept 'Checking for errors' if not qualified with 'not picked up in testing'.	**(1)**

Question number	Answer	Additional guidance	Mark
4(c)	A linked description that makes reference to the **four** following points: • The address of memory (holding instruction) is placed on the address bus (1). • The control unit sends a signal (1) on the control bus (to start a read operation) (1). • The instruction is/the contents of the memory are placed on the data bus (1).		**(4)**

Question number	Answer	Additional guidance	Mark
4(d)	A linked description that makes reference to any **four** of the following points: • All processes are held in a queue (1) • Processes are prioritised (1) • Processes are allocated time slices (1) • Length of time slice depends on priority (1) • (and) processes are switched (at the end of their time slice) (1) • Unfinished processes are put to the back of the queue (1) • During the time slice the process has exclusive use of the processor (1).		**(4)**

Question number	Indicative content	Mark
4(e)	Advantages of high-level languages: • High-level languages come with libraries of ready-made graphical user interface components (buttons, icons and menus), which the team can use to reduce the amount of code they have to write from scratch. • High-level languages have a range of integrated development tools, editors and syntax checkers, which will enable the team to develop the interface code more efficiently. • Portability is a real consideration: should the company decide to use a different chipset in the future, programs written in a high-level language won't need to be rewritten. They can be recompiled to run on new architecture relatively quickly. • High-level languages use keywords, which will enable team members to read and understand the code for the user interface more easily. • There are lots of people who can program in a high-level language, making it relatively easy to recruit experienced programmers to the user interface team. • As high-level language translators exist for a range of operating systems, each member of the user interface team can develop code in their preferred environment. Advantages of low-level languages: • Code written in assembly language normally executes more quickly and takes up less memory than code written in a high-level language. This may be crucial to enable the control unit for the alarm system to function effectively. • There may be no high-level language for the microprocessor chip inside the control unit, so an assembly language would have to be used for it. • Code written in assembly language allows the programmer to directly control system hardware.	**(6)**

Level	Mark	Descriptor
	0	No rewardable content.
Level 1	1–2	Basic, independent points are made, showing elements of understanding of key concepts/principles of computer science. (AO1) The discussion will contain basic information with little linkage between points made or application to the context. (AO2)
Level 2	3–4	Demonstrates adequate understanding of key concepts/principles of computer science. (AO1) The discussion shows some linkages and lines of reasoning with some structure and application to the context. (AO2)
Level 3	5–6	Demonstrates comprehensive understanding of key concepts/principles of computer science to support the discussion being presented. (AO1) The discussion is well developed, with sustained lines of reasoning that are coherent and logically structured, and which clearly apply to the context. (AO2)

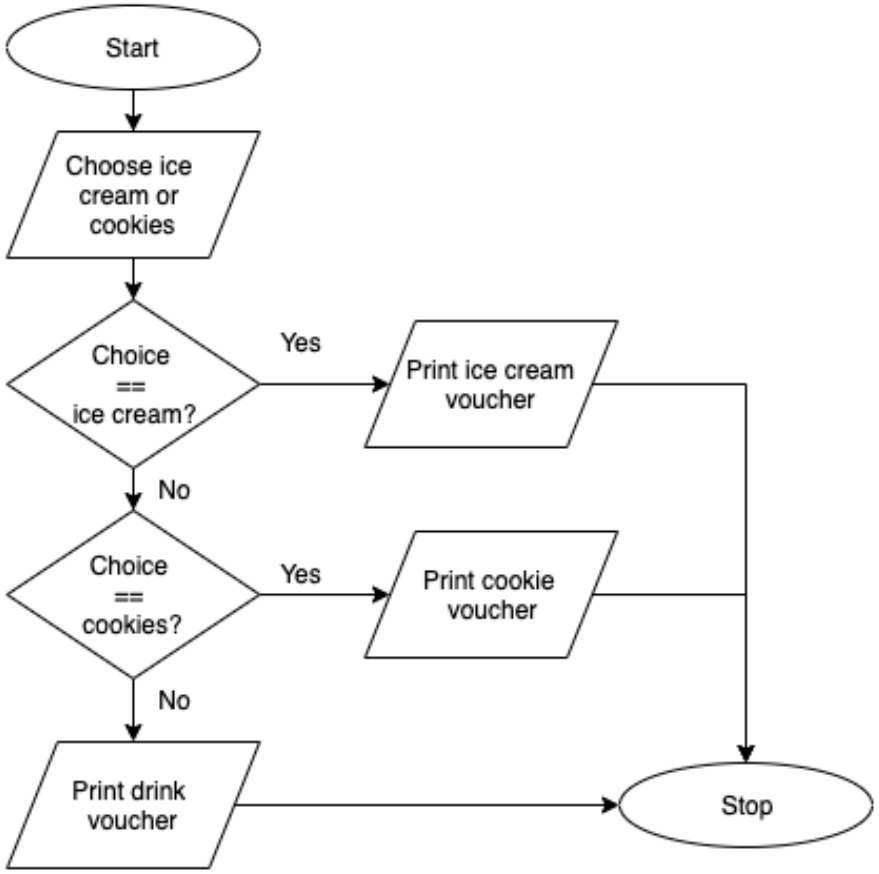
Question number	Answer	Additional guidance	Mark
5(a)	Logic		**(1)**

Question number	Answer	Additional guidance	Mark
5(b)	Award **1** mark for each correct cell. First: f Second: c Third: a Fourth: b		**(3)**

Question number	Answer	Additional guidance	Mark
5(c)	Award **1** mark for the identification of a reason (1) with a linked justification/exemplification (1), up to a maximum of 2 marks. • Constants (shown in all capitals) are less likely to be changed by accident or error (1), so algorithms that use them should be more robust (1). • If the value of a constant does have to be altered (1), only one change is required (on the line where it is created and set) (1). • Constants allow values to be replaced with a name/identifier (1), so code is easier to read/maintain (1).		**(2)**

<table>
<tr><th>Question number</th><th>Answer</th><th>Additional guidance</th><th>Mark</th></tr>
<tr><td>5(d)</td><td>1 mark for initialising all variables and 1 mark for each correct pass through the loop.

<table>
<tr><th>num</th><th>x</th><th>y</th><th>Display</th><th>Marks</th></tr>
<tr><td>0</td><td>999</td><td>0</td><td></td><td>(1)</td></tr>
<tr><td>355</td><td>355</td><td>355</td><td></td><td>(1)</td></tr>
<tr><td>554</td><td></td><td>554</td><td></td><td>(1)</td></tr>
<tr><td>199</td><td>199</td><td></td><td></td><td>(1)</td></tr>
<tr><td>409</td><td></td><td></td><td></td><td>(1)</td></tr>
<tr><td></td><td></td><td></td><td>199 554</td><td>(1)</td></tr>
</table></td><td>• Award alternative versions of the trace table if correct. For example, copying of values that do not change.
• Passes are incorrect if display is indicated.
• Display must be after the final pass (on a separate line in the table).</td><td>(6)</td></tr>
</table>

Question number	Answer	Additional guidance	Mark
5(e)	• Correct message in output box acting as a prompt for the user (1). • Correct diamond symbol for decision (1). • Correct test 'Choice == cookies?' for decision (1). • Correct label 'Yes' on right arrow AND Correct label 'No' on bottom arrow (1). • Correct output symbol with suitable message (1). • Correct ellipse symbol and 'stop' for terminator (1). 	• Symbol and contents are awarded independently. • Award 'End', 'Stop', 'Start' and 'Begin' as text for terminator symbols. • Award '==' and '=' used for equivalence inside decision symbol, but not in process symbol. • Accept 'Input choice' as an alternative response in the top process symbol	**(6)**