

School Of Coding

**GCSE OCR (9-1) Computer Science**

**Component 1 - Computer Systems &**

**Component 2 - Computational Thinking, Algorithms and Programming**

Practise MIxed Paper

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Question** | | | **Answer** | **Mark** |
| 1 | a |  | 1 mark per bullet point:   * A computer uses RAM to store programs (and data).... * while they are being executed - condone being run / processed | 2 |
| 1 | b |  | 1 mark per bullet point:   * A computer with 8 gigabytes of RAM can have more programs opened at the same time. * all the programs might fit in RAM and not have to be read in and out to/from disc | 2 |
| 1 | c |  | 1 mark per bullet point:   * It has four processors / * carry out four instructions at the same time * It means the computer could run programs faster | 2 |
| 1 | d |  | 1 mark per bullet point:   * This is the clock speed of the processor 8GHz means (8 billion or 8 000 000 000) ... * cycles per second | 2 |
| 2 | a |  | 2 marks for description of each layer  E.g  Application layer   * Allows the browser software to communicate [1] * with the world wide web via a protocol [1] * such as HTTP or HTTPS. [1]   Transport Layer   * Determines the route for the transmission [1] * and opens up socket connections [1] * using TCP [1] * which is reliable protocol [1] * Splits the data into sequentially numbered packets [1]   Internet Layer   * Adds IP address headers [1] * to the transmission and routes packets to their destination [1]   Link Layer   * Adds the MAC address of the network device to the packet [1] * and physically transmits the data along the medium being used [1] | 8 |
| 2 | b |  | * POP, [1] * IMAP [1] * SMTP [1] | 3 |
| 2 | c |  | 1 mark for each bullet point, upto 6 marks.   * POP is used for receiving an email * IMAP is used for receiving an email * SMTP is used when sending an email * POP downloads the email from the server whereas IMAP reads it online * POP deletes email from the server whereas IMAP leaves the mail on the Server. * IMAP syncs the emails on all devices while POP does not | 6 |
| 3 | a |  | 1 mark for what it provides, 4 marks for description based on the situation   * A mesh network can provide emergency coverage. [1]. * They don’t rely on any specific infrastructure [1] …. * so can be set up quickly [1]. * Easily extended into areas where there is not normally any coverage [1]... * by using portable technology [1]. | 5 |
| 3 | b |  | 1 mark for an advantage, 1 mark for a disadvantage   * Disadvantages:   + Every computer sends/forwards all the data , meaning the available bandwidth is lower ...   + Whereas in a star network, the data is only switched/sent to the target machine   + Difficult to manage/maintain   + Lack of centralised control/services ...   + Whereas a star network has a centralised server and set of resources * Advantages:   + Very robust and reliable   + Not reliant on a server   + If one computer breaks the rest are fine. | 2 |
| 4 | a | i | 30 | 1 |
| 4 | a | ii | 45 | 1 |
| 4 | b |  | 1 mark for each bullet point   * Reject inappropriate data // only accept appropriate data * To ensure the program will not crush / cause problems later on from the values input * To ensure the program handle invalid / inappropriate data correctly | 2 |
| 4 | c |  | * Repeatedly asking for the input (e.g in a loop) [1] * Looping until *username == “playerA”* …. [1] * Printing that the username is wrong [1]   e.g  username = input(“Please enter your username again.”) while username != “playerA” then  print(“Your username is wrong. Try Again”)  username = input(“Please enter your username again.”) | 3 |
| 4 | d |  | * Opening *resultsData.txt* in write mode [1] * ….using a variable [1] * Writing final to file * Closing the file   e.g file = OPENWRITE( “resultsData.txt”) file.WRITELINE(final) file.CLOSE() |  |
| 5 | a |  | 1 mark for description, 1 mark for example   * High-Level is in English sentences/words/phrases [1] * Low-level is binary [1] * High-level is written/understood by programmers [1] * Low-level is understood/executed by a processor // does not need translating [1] * High-level needs translating (into low-level) before it can be run. [1] | 4 |
| 5 | b |  | 1 point for naming a feature for each , 1 point for describing it  Interpreter   * Feature: runs/checks the code line by line * Expansion: debugging / easier to find errors   Compiler   * Feature: produces executable files / .exe * Expansion: Can give program without source code * Expansion: Can run/test without recompiling/ retranslating * Feature: checks all code at once * Expansion: Can see/correct errors before testing. | 4 |
| 5 | c | i | 1 mark for each bullet point, upto 3 marks.   * You can write/edit/read code * Colour coding for key words * Auto-complete * Auto-indent | 3 |
| 5 | c | ii | 1 mark for each bullet point, upto 2 marks.   * Allows the code to be run/executed * View the results of the execution * Allows input of data | 2 |
| 5 | d |  | * Indenting * Show where constructs start and end * Comments * Explain what lines/constructs/procedures do * Appropriate identifier names * Use names that reflect the meaning/ purpose |  |
| 6 | a |  | 1 mark for each bullet point   * Create a sorted list and an unsorted list * Take each item in the unsorted list * … and place it in the correct position in the sorted list * The sorted list expands until there are no elements left in the unsorted list * Moving 8 to the correct place * Moving 4 to the correct place * Moving 2 to the correct place   e.g | 6 |
| 6 | b |  | 1 mark for each bullet point   * Compare elements in pairs from start-end/left-to-right * Swap elements if they are in the wrong order * When you reach the end of the array/list/items start again * Continue until you move through the entire list/elements without making any changes   e.g | 6 |
| 5 | c |  | Merge Sort | 1 |
| 7 | a | i | 00110001 | 1 |
| 7 | a | ii | 01101010 | 1 |
| 7 | a | iii | 43 | 1 |
| 7 | a | iv | 173 | 1 |
| 7 | b |  | 1 mark for getting a correct answer, 1 mark for showing workings.  4x1024x1024 = 4194304 /  4x1000x1000 = 4000000 | 2 |