GCSE 9-1

Computer Science

Mark Scheme for Mixed Paper 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Question** | | | **Answer** | **Mark** | **Guidance** |
| 1 | a |  | 1 mark ber bullet, max 2   * Faster transmission speeds * Connections not obstructed by walls * More secure (as a physical connection is needed) * Max distance for reliable communication is longer * Higher bandwidth * No interference * Not very mobile | 2 |  |
| 1 | b |  | 1 mark ber bullet, max 2   * Easier setup * Easy to connect devices into network * Cheap (do not need to purchase wires, only WAP) * Very mobile | 2 |  |
| 1 | c | i | * Covers a small/local geographical area * Uses internal/has its own infrastructure/cables/hardware for connections | 2 |  |
| 1 | c | ii | * Covers a large geographical area * Uses external hardware/infrastructure/cables for connections | 2 |  |
| 1 | d |  | 1 mark for reason, 2 marks for explanation   * No. of devices * ...large number of devices reduces performance * ...as devices must share bandwidth * ...more devices means higher chance of collisions * Bandwidth * ...smaller bandwidth, less data can be transferred per second/unit of time * ...smaller bandwidth reduces performance * Hardware used * ...fibre optic wires have higher bandwidth/can transmit data faster * ...using fibre optic wires increases performance * ...hub-based network has higher latency than switch-based network * ...so using a hub results in poorer performance * Latency * ...higher latency means larger delay between when data is sent and received * ...higher latency results in poorer performance * Error rate * ...more errors means data has to be resent, taking time * ... higher error rate reduces performance * Interference * ...thick walls can result in weak Wi-Fi connection * ...interference from devices with an electro-magnetic signal can cause dropped connections * ...high levels of interference result in poorer performance | 6 |  |
| 2 | a |  | 2 marks per component, 1 for naming, 1 for explaining its purpose   * Control Unit * ...decodes instructions * ...issues control signals to hardware * ...moves data around system * ...controls execution of instructions in correct sequence * ...contains system clock which controls processor timing using regular pulses * Cache * ...stores frequently used instructions/programs/data * ...increases access speed improving performance * ...data does not need to be fetched from RAM | 2 |  |
| 2 | b |  | FULL name of register needs to be entered  1 mark per correct name   |  | | --- | | **Name of Register** | | Memory Address Register | | Memory Data Register | | Program Counter | | Accumulator | | 4 |  |
| 2 | c | i | 1 mark per bullet, max 2   * Small computer system that forms part of a larger system * Has a limited number of tasks | 2 |  |
|  |  |  | * Any suitable example, e.g. dishwasher, microwave | 1 |  |
| 3 | a |  | * Non-volatile * Contains BIOS/bootstrap | 2 |  |
| 3 | b | i | * 8 \* 1000 \* 1000 * 8,000,000 Bytes | 2 |  |
| 3 | c | i | * Solid state box ticked | 1 |  |
| 3 | c | ii | 1 mark for benefit of solid state, 1 mark for linking to situation   * Faster access speeds * ...his projects will load/open faster * Very portable/small physical size * ...can transport files/projects very easily * Very durable * ...projects will not be lost easily * ...no moving parts * Not sensitive to being moved * ...Ruben can transport his projects easily * Reliable * ...Low chance of his projects being lost accidentally * Sufficient/large capacity * ...can store many projects without needing multiple USBs | 4 |  |
| 3 | c | ii |  |  |  |
| 4 | a |  | 2 marks max per process  Memory Management   * Puts applications that are in use/to be used into RAM/Main Memory * Takes programs not being used out of RAM * Moves programs to and from Virtual Memory   Multitasking   * Allows more than one program to (appear to) run at the same time * Shares processor time between tasks * Splits processor time into “time slices” * Allocates time slices to each process | 4 |  |
| 4 | b | i | * Compression is reducing the size of a file   Needed to: (1 mark max for items below)   * Allow files to be uploaded to the internet faster * Allow files to be downloaded from the internet faster * Maximise storage space in a computer/storage device * Allow files to be sent as attachments in emails (or other services) | 2 |  |
| 4 | b | i | * Lossy compression permanently deletes data * ...whereas lossless compression doesn’t delete any data * ...whereas lossless compression uses algorithms to encode data and store differently * With lossy compression, the file can never revert back to the original * ...whereas with lossless compression, the file can be turned back into the original file * Lossy compression reduces file size more * ...whereas lossless compression is less effective at reducing file size | 2 |  |
| 4 | b | iii | 1 mark for type of compression, 2 marks for explanation   * Lossy   + Achieves higher compression/smaller file size/faster streaming than lossless   + Sound can still be listened to/understood at a lower quality | 3 |  |
| 4 | b | iv | 1 mark for type of compression, 2 marks for explanation   * Lossless   + Book must be exactly as originally written with no data removed   + Otherwise the book cannot be understood/is unintelligible. | 3 |  |
|  |  |  |  |  |  |
| 5 | a |  | * Each character is given a numeric code * This code is then stored in binary * Characters can take 7 bits, 8 bits or 16 bits depending on character set * Text is stored as series of binary numbers | 2 |  |
| 5 | b |  | * 100 0010 (1 added to code for A) | 1 |  |
| 5 | c |  | * 100 0100 (3 added to code for A) | 1 |  |
| 5 | d |  | Allow use of numbers given in parts b and c  Must be binary code for D then binary code for A then binary code for B in order  One mark for each correct letter, must be in order   * 1000100 1000001 1000010 | 3 |  |
| 5 | e | i | * Unicode * Unicode uses 16 bits to represent each character * Can represent a very wide range of characters including emojis * Can represent 2^16 characters | 3 | Need to give the name of character set or do not award marks for explanation |
| 5 | e | ii | 1 mark per bullet to max 3   * ASCII * Uses 7 bits to represent each character * Do not need more than 7 bits to represent each character * Represents english letters and numbers and basic symbols. * Represents all characters needed in an english children's book * Using a different character set is unnecessary | 3 | Need to give the name of character set or do not award marks for explanation |
| 5 | f |  | * Will divide it * By 4 (by 2 x 2) (by 2^2) | 2 |  |
| 5 | g |  | * 00011010 | 1 |  |
| 5 | h |  | Give 2 marks if answer given   * 16 + 8 + 2 * 26 | 2 | Allow use of the answer from part f |
| 6 | a |  | 2 mark per bullet, max 2   * It is a parameter * It is a variable used in a subroutine/function/procedure * Allows us to pass values/data into a subroutine/function/procedure * Acts as a placeholder for data entered when the function is called | 2 |  |
| 6 | b |  | 1 mark per bullet, max 2   * Breaks down / decomposes / modularises the problem / program // structures the program * …making it easier to design/create/test * …each subroutine can be tested separately * Reuse code (in different programs) * …quicker to develop (new) programs * …build on existing work / use of a library of subroutines * Avoid repetition of code (in the same program) * …makes program shorter / smaller * … subprogram called instead of copying/pasting. * … quicker to develop (new) programs * Easier to maintain …as code is easier to understand/read * …as code is shorter * Easier to debug * …as code is shorter * …same bugs will not have been copied to other areas of the program. * Work can be split up in a team * ...to suit developers’ skill set * …to work on different subprogram at the same time / develop separately * Allows for abstraction / removes complexity * …subprograms can be used by programmers who do not need to understand how they work. | 2 | Allow “can be called multiple times”  Allow “file size is smaller”.  Do not allow “more efficient” without further explanation. |
| 6 | c |  | |  |  |  | | --- | --- | --- | | **size** | **result** | **noOfItems** | | 0 |  | 0 | | 8 | even | 1 | | 11 | odd | 1 | | -1 |  | 1 | | 4 |  |
| 6 | d |  | * Creates a 2 dimensional array that stores 5 items across and 2 items down (or 2 items across and 5 items down) * Asks the user for the barcode ID and stores it in an appropriately named variable * Asks the user for the season of the item of clothing and stores it in an appropriately named variable * Adds the barcode ID to the array * Adds the season to the array * Adds these items to the array so that they are held “together” * Repeats bullets 2,3,4,5 and 6 * ...4 more times (5 times total) | 8 | Example program:  array clothing [4,1]  for x = 0 to 4  barcodeID = input("Enter barcode ID")  season = input("Enter season")  clothing[x,0] = barcodeID  clothing[x,1] = season  next x |

**School of Coding**

8 Newton Court

Pendeford Business Park

Pendeford

Wolverhampton

WV9 5HB

**Contact Details:**

Telephone: 01902 509 209

Email: info@schoolofcoding.co.uk

[www.schoolofcodinguk.com](http://www.schoolofcodinguk.com)

