|  |  |  |
| --- | --- | --- |
| Question | Description | Marks |
| 1 (a) | One mark for each correct definition:* The sample rate is the number of samples taken in a second/per time unit
* The sample resolution is the number of bits per sample
 | 2 |
| 1 (b) | * Lossy compression
 | 1 |
| 1 (c)(i)  | Any two from:e.g.* Destination/receivers (IP) address
* Packet number
* Originator’s/senders (IP) address
 | 2 |
| 1 (c)(ii) | Any five from:* Data is broken/split/divided into packets
* Each packet (could) take a different route
* A router controls the route/path a packet takes
* … selecting the shortest/fastest available route/path
* Packets may arrive out of order
* Once the last packet has arrived, packets are reordered
* If a packet is missing/corrupted, it is requested again
 | 5 |
| 2 (a) | One mark per each correct character in the correct order:* 9
* 3
* 0
* D
 | 4 |
| 2 (b)(i) | * 00001111
 | 1 |
| 2 (b)(ii) | Any one from:* The value becomes incorrect/inaccurate as the right most bits are lost
* It is divided by 8
 | 1 |
| 2 (c) | Any two from:* Easier/quicker to understand/read/write
* Easier/quicker to debug
* Less likely to make a mistake
* Shorter representation // Takes up less screen space
 | 2 |
| 2 (d) | One mark for two correct characters, two marks for three correct characters in the correct order:* 1
* 2
* D
 | 2 |
| 3 (a) | Any three from:* A character set is used
* … such as Unicode/ASCII
* Each character has a unique binary value
 | 3 |
| 3 (b)(i) | * It reduces the file size
 | 1 |
| 3 (b)(ii) | Any four from: • A compression algorithm is used • … such as RLE/run length encoding • Repeating words/characters/phrases are identified // Patterns are identified • … and indexed • … with number of occurrences • … with their position | 4 |
| 3 (b)(iii) | Any two from: e.g. • To save storage space • To make it quicker to transmit • To make it small enough to attach to an email • To reduce the bandwidth needed to transmit | 2 |
| 4 (a)(i) | Two from: • Data is sent one bit at a time • A single wire is used | 2 |
| 4 (a)(ii) | Any two from: • Data won’t be skewed • Less chance of interference/crosstalk/corruption/error • Transmission speed is adequate | 2 |
| 4 (a)(iii) | * The data may be transmitted quicker
 | 1 |
| 5 (a) | * Each correct line (1 mark)

“S1110”REAL54INTEGER“TRUE”BOOLEANFalseSTRING60.25“12/12/2000” | 6 |
| 5 (b)(i) | Any one:* a diagram of the sequence of movements or actions or activity.
* a graphical representation of a computer program
 | 1 |
| 5 (b)(ii) | Any one:* description of the steps in an algorithm
* detailed description of what a computer program or algorithm should do written in a formal.
 | 1 |
| 5 (b)(iii) | * Storage location that can change its value during program
 | 1 |
| 5 (b)(iv) | A value that does not change over time | 1 |
| 6 | One mark per bullet point55• Data type name Integer• Data type description (Any) whole numberBS1100• Data type name String• Data type description A group of characters/text20.0• Data type name Real• Data type description (Any real) number that could be a whole number or a fractionTrue• Data type name Boolean• Data type description True and False“False”• Data type name String• Data type description A group of characters/text | 10 |
| 7 (a) | Variable declaration 1 marksinput 1 marksSelection statement 3 marksOutput 1 marksDeclare num1, num2, num3 : INTEGERINPUT “Enter First Number “, num1INPUT “Enter second Number “, num2INPUT “Enter third Number “, num3IF num1 > num2 AND num1 > num3 THEN OUTPUT num1 , “is greater” ELSE IF num2 >= num1 AND num2 > num3 THEN OUTPUT num2 , “is greater” ELSE IF num3 > num1 AND num3 > num2 THEN OUTPUT num3 , “ is greater” | 6 |
| 7 (b) | 1 mark for each circleStartINPUT Num1, num2, num3 ATrueIsNum1 > num2AND num1 > num3?OUTPUTNum1, “is greater AOUTPUTNum2, “is greater AIsNum1 > num2AND num1 > num3?TrueIsNum1 > num2AND num1 > num3?TrueOUTPUTNum3, “is greater AEND | 8 |
| 8. a | FOR initialization TO condition Statement(s)NEXT | 2 |
| 8. b | IF (condition) // statement(s) if TRUEELSE // statement(s) if FALSE | 2 |
| 8. c | WHILE (condition) DO // statement(s)  // increment / decrementENDWHILE | 2 |