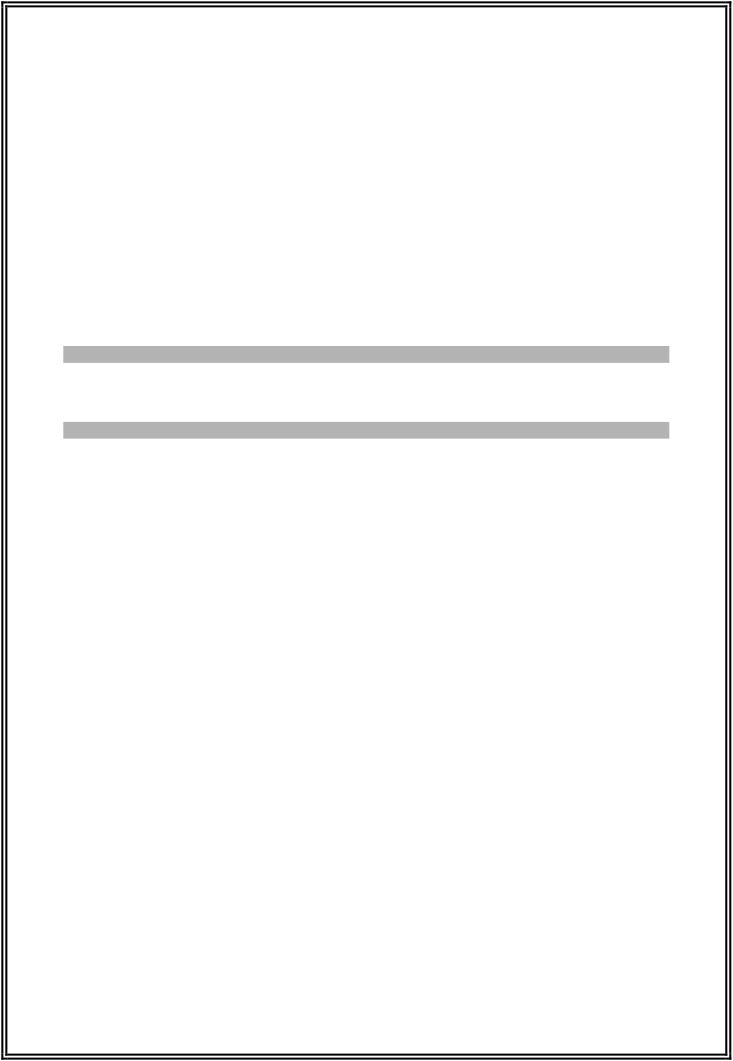
**Evaluation Only. Created with Aspose.Words. Copyright 2003-2023 Aspose Pty Ltd.**



# **GCE AS MARKING SCHEME**



**SUMMER 2019**

**AS (NEW)**

**COMPUTER SCIENCE - COMPONENT 1 B500U10-1**

# **INTRODUCTION**

This marking scheme was used by WJEC for the 2019 examination. It was finalised after detailed discussion at examiners' conferences by all the examiners involved in the assessment. The conference was held shortly after the paper was taken so that reference could be made to the full range of candidates' responses, with photocopied scripts forming the basis of discussion. The aim of the conference was to ensure that the marking scheme was interpreted and applied in the same way by all examiners.

It is hoped that this information will be of assistance to centres but it is recognised at the same time that, without the benefit of participation in the examiners' conference, teachers may have different views on certain matters of detail or interpretation.

WJEC regrets that it cannot enter into any discussion or correspondence about this marking scheme.

**GCE AS COMPUTER SCIENCE SUMMER 2019 MARK SCHEME**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Q** | **Answer** | **Marks** | **AO1** | **AO2** | **AO3** | **Tot** |
| 1(a) | Award one mark for each of the following:   * Half-duplex * Data can travel in both directions but only in one direction at any given time | 1 1 |  | 2.1b 2.1b |  | 2 |
| 1(b) | Award one mark for each of the following:   * Simplex * Data can only travel in one direction | 1 1 |  | 2.1b 2.1b |  | 2 |
| 1(c) | Award one mark for each of the following:   * Full duplex * Data can travel in both directions at the same time | 1 1 |  | 2.1b 2.1b |  | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Q** | **Answer** | **Marks** | **AO1** | **AO2** | **AO3** | **Tot** |
| 2 | Award one mark for each name and one for each description of the following up to a maximum of eight (4x2)  CIR   * Register that holds the instruction currently being executed.   Registers   * A small amount of fast access storage * Normally used for a specific purpose where data or control information is temporarily stored. * Control Unit * Directs the flow of instructions and/or data * Coordinates the other parts of the CPU * Generates clock ticks or controls the clock * Arithmetic Logic Unit * The ALU performs all the mathematical calculations and logical operations in the   CPU.   * MDR * Register of a computer's control unit that contains the data to be stored in the computer storage (e.g. RAM), or the data after a fetch from the computer storage. * MAR * Register that either stores the memory address from which data will be fetched to the CPU or the address to which data will be sent and stored. In other words, MAR holds the memory location of data that needs to be accessed. * PC * Processor register that indicates where a computer is in its program sequence. * Buses * Connects all the internal components of a computer, such as CPU and memory, to the motherboard. * Cache memory * stores copies of the data from frequently used main memory locations. * Most CPUs have different independent caches, including instruction and data caches, where the data cache is usually organized as a hierarchy of more cache levels (L1, L2, etc.) | 8 | 1.1b |  |  | 8 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Q** | **Answer** | **Marks** | **AO1** | **AO2** | **AO3** | **Tot** |
| 3 | Award one mark for each of the following up to a maximum of two marks for each method:   * File backup   + A data backup is a copy or archive of files and folders for the purpose of being able to restore them in case of data loss. * This involves storage of three of the most recent versions of master file. (grandfather –   father - son)   * Useful if one version is corrupted: the previous version(s) is still available. * Data should be stored off site in case of a disaster.   Access Rights   * Users can be given rights to certain files or file structures that prevent them from accessing them / changing them / deleting them. * File attributes   Firewall   * Prevents unauthorised access   Encryption   * Users are unable to access data within a file without the correct key / decryption   Passwords   * Users are unable to access data within a file without the correct password | 2  2 2 2 2 | 1.1b  1.1b 1.1b 1.1b 1.1b |  |  | 6 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Q** | **Answer** | **Marks** | **AO1** | **AO2** | **AO3** | **Tot** |
| 4(a)(i) | **One mark for each correct answer up to a maximum of three**   * Relational database * Uses a primary key in each table which is a unique identifier, such as Customer ID in the Customers Table * Uses a foreign key, which is a primary key from another table, forming a link between the tables. For example, Customer ID in the Bookings Table. * 1:M relationships are used to link tables | 1 1  1 1 |  | 2.1b 2.1b  2.1b 2.1b |  | 3 |
| 4(a)(ii) | **Award one mark for each of the following up to a maximum of two.**  Benefit:   * Avoids data duplication * Avoids inconsistent records * Saves storage space   **Award one mark for each of the following up to a maximum of two.**  Drawback:   * Data Complexity when data resides in multiple tables, which are linked to each other through shared key values. * Maintaining / modifying relational databases can be difficult * Broken keys and records producing errors in other tables / queries | 2  2 |  | 2.1b  2.1b |  | 4 |
| 4(b) | * Database view allows a number of tables and records to be **restricted** so only certain users can see / amend certain sets of data.   **Award one mark for any of the following:**   * For example, certain staff should not be able to see / amend salary information, but someone from management could.   Other examples accepted where the restriction of sensitive data is illustrated. | 1  1 |  | 2.1b  2.1b |  | 2 |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Q** | **Answer** | **Marks** | **AO1** | **AO2** | **AO3** | **Tot** |
| 5 | **Indicative content**  1 Declare PTArray[18,7] **(accept 17,6)**  2 Group is integer  3 Period is integer  4 Element is string  5  6 input Period  7 input Group  8  9 if PTArray [Group, Period] = Null then 10  **(accept reverse order indexing)**  11 Output “Error, not found”  12 else  13 Element = PTArray[Group, Period] 14 Output "Element is:” , Element  15 end if  Award one mark for each of the following:   * Declare / initialise variables * Input Period * Input Group * Look up for Error (blank, null, error, zero) |

**This document was truncated here because it was created in the Evaluation Mode.**