GCSE EXAMINERS' REPORTS

GCSE
COMPUTER SCIENCE

SUMMER 2023

Grade boundary information for this subject is available on the WJEC public website at: https://www.wjecservices.co.uk/MarkToUMS/default.aspx?l=en

Online Results Analysis

WJEC provides information to examination centres via the WJEC secure website. This is restricted to centre staff only. Access is granted to centre staff by the Examinations Officer at the centre.

Annual Statistical Report

The annual Statistical Report (issued in the second half of the Autumn Term) gives overall outcomes of all examinations administered by WJEC.

COMPUTER SCIENCE

GCSE

Summer 2023

UNDERSTANDING COMPUTER SCIEMCE

General Comments

The mean total for unit 1 this series was approximately 38 out of 100 marks. This represents a slight decrease when compared with the mean total of 41 out of 100 in 2022 but is very close to the mean pre-covid in 2019.

The effect of the pandemic remains evident in pupils' literacy skills and their use of technical terminology, but less so than in 2022.

Comments on individual questions/sections

Q.1 Many candidates were able to identify if statements about a CPU were true or false. Very few were able to use accurate technical terminology when describing the fetch-decode-execute cycle. Around half of candidates were able to correctly describe the difference between an embedded system and a general-purpose computer system.

Q.2 Very few candidates were able to explain why flash memory would be better to store the system BIOS than ROM. Many were able to give at least one benefit to increasing the size of RAM, but only a few candidates were able to give two correct benefits. Most candidates were able to name two input and two output devices. This question was the second best answered question of the paper.

Q.3 Around half of candidates were able to complete the given truth table. These candidates were also able to simplify the Boolean expression. Candidates tended to get full marks or zero marks for the Boolean simplification.

Q.4 This question was the worst answered question of the paper. Around half of candidates were able to identify the correct protocol for the given description. Very few candidates were able to correctly state the role of both the TCP and IP protocols in network communication. Many candidates were able to name at least two other items found in a TCP/IP packet, but fewer were able to name three. Around half of candidates were able to correctly give one advantage and one disadvantage of circuit switching. Very few candidates were able to explain how DNS resolves addresses when a user attempts to visit a website. This last part of the question in particular was poorly answered, by most candidates.

Q.5 This was the best answered question of the paper, with many candidates determining the lowest cost routes between each origin and destination.

Q.6 Many candidates were confident in converting between number counting systems. Very few candidates were able to describe the process of carrying out arithmetic shifts to the left and to the right, instead they discussed the doubling and halving effect. Around half of candidates were able to use binary addition to add two numbers, but a few lost a mark for the lack of carry bit.

Q.7 A few candidates were able to complete the given data structure table, by firstly choosing an appropriate field name and then matching it with a suitable data type and method of validation. Where candidates lost marks, this was due to the data type or validation not matching the field. Many candidates were able to describe two software methods that could be used to protect the security of data. A few were able to use XOR to demonstrate how data could be encrypted.

Q.8 This was the third best answered question of the paper. Around half of candidates were able to complete the sentences about the different IDE tools.

Q.9 Around half of candidates were able to identify the errors in the program and suggest a suitable change to the code to address each error. The majority of candidates were able to give two benefits to programmers of using a high-level programming language, but only a few were able to name two stages of the compilation process.

Q.10 This was the joint-second worst answered question of the paper. The majority of candidates were able to give two principles of the Computer Misuse Act which help businesses protect their data. However, very few were able to describe three different types of targeted attack against confidential data. Note that malware, viruses and hacking was deemed too generic for this question. A few were able to describe two methods of identifying vulnerabilities.

Q.11 This was the joint-second worst answered question of the paper. A few candidates were able to describe the purpose of different utility software provided by the operating system and the functionality of the operating system in managing resources. These candidates attempted to address the question but demonstrated superficial knowledge of the indicative content. They addressed the question by discussing only utility software or operating system resources. They also used limited technical terminology referring to the indicative content. It is worth noting that a significant number of candidates (nearly a quarter) did not attempt this question.

Summary of key points

Candidates had a reasonably good understanding of primary storage, lowest cost routes and IDE tools. They performed best in these questions.

Candidates found the questions on protocols, circuit switching and DNS, Computer Misuse Act and cyber-security, and operating systems difficult. This was reflected in the low facility factor for these questions.

COMPUTER SCIENCE

GCSE

Summer 2023

COMPUTATIONAL THINKING AND PROGRAMMING

General Comments

Most candidates demonstrated a good understanding of the specification. Many candidates were well prepared, and many excellent answers were evident. There was evidence also that most candidates had been well prepared for some of the practical programming elements. It was pleasing to see the Java programming questions answered so well. With programmed screen savers and interactive games in evidence.

Comments on individual questions/sections

HTML (markup language):
This was answered well by most candidates. However, a significant number of candidates found the link (using <a href=. . .) difficult. Candidates also found the need to use <ul> and <li> tag difficult. As standard, candidates also forgot to close the HTML tags resulting in incorrect formatting. Candidates should not use generated code or CSS as the specification states the HTML tags that should be used. Some candidates confused the Head, Title and Body tags.

Assembly language:
This is considered to be a more difficult area of the specification however item level data pleasingly reported that this was well attempted and shows an improvement in understanding by candidates.

Algorithms:
Some good answers were seen to this question, however, only a few candidates achieved full marks. Sorting algorithms were found to be more difficult and the item level data shows this is the most demanding topic on the paper.

Java programming within the Greenfoot environment:

Populating the world proved problematic for a very small number of candidates, most candidates could get an object to move around the screen. A significant number could get an object to respond to key input. Random movement was programmed well, and many candidates also had object collision detection in evidence. Question 5 (The screen saver) had the highest facility factor and was well attempted by the majority of candidates.

GCSE

Summer 2023

SOFTWARE DEVELOPMENT

General Comments

Examples of good work were seen during moderation this summer. This specification requires candidates to undertake the one scenario that is made available to candidates. It should be noted that it is essential that candidates undertake the correct scenario for the current series.

Comments on individual questions/sections

The scenario has a bullet pointed lists of requirements. To access full marks for the implementation of the solution to the given problem, all bullet points should be covered. However, many candidates were not able to produce a solution that covered all bullet points of the scenario.

The scenario was based on the creation of a loyalty scheme system for the shops in Parkwood Vale. The scheme required candidates to create software that allowed for sale of SIP ('Shop In Parkwood Vale') cards which allowed customers to access specific deals from local business, i.e.:

- A discount of 10% on certain days of the week
- Free delivery of orders
- Two for one on selected products

There was a charge of £2.00 for access to the scheme and the card could be purchased from any of the local businesses. The card would be valid for a period of a calendar year from the point of purchase.

The system would be required to record customer purchases per shop and note which discount was utilised, if the card was valid (in date). A final monthly report would need to be generated that noted all claimed discounts by customers from a particular business.

Most candidates could create an interface that allowed users to enter the customer's personal details when purchasing the SIP card. Many candidates were able to produce code that could carry out the required functions. Candidates also created validation routines. However, these routines often caused issues when running the code as the interfaces did not provide clear enough instructions for the end user.

Requirements for the Report:

The specification states that the candidates should produce a report that:

- analyses the given information
- includes a design of a solution to the given problem
- programming of a solution to the given problem
- testing and refinement of the application, noting the refinements in the refinement log
- gives an evaluation of the application

Refinement log

Candidates are required to complete a log of their activities during the twenty-hour controlled assignment. Almost all candidates presented a completed log. However, a significant number of candidates submitted logs that included many copied and pasted entries. Where candidates had made effective use of their logs; entries included discussion of problems encountered and solutions to these problems. Many candidates were able to identify action points for following sessions that would enable them to make more effective use of their time.

Scope of the problem

A minority of candidates presented effective analysis of the given scenario while many either restated the problem or copied and pasted the contents of the brief. Most candidates were able to outline the objectives for their solution to the given problem.

Design of Solution

In a significant minority of instances, candidates were neither able to justify their choice of programming language using appropriate technical terminology nor relate the features and facilities of the language to their proposed method of solution.

Many candidates were able to describe some of the process stages required for their solutions in pseudo code and/or flowcharts. However, fewer candidates covered all processing stages for their proposed solution. In several cases it was not clear that this work had been completed before implementation. Retrospective designs will not be given any credit at moderation.

Effectiveness of solution

Most candidates had produced a solution that allowed the user to enter the details of the businesses and customers into the system. A large percentage of candidates developed solutions using the Python programming language, while others produced solutions using a visual programming language. The quality of the interfaces produced varied considerably. Centres are reminded that the use of online software development tools is prohibited and that ‘drag and drop’ block type IDEs is not adequate.

Most solutions were modular and included the required authentication routines.

Technical quality

Many candidates produced code that was self-documenting and there were instances of the code being well structured. Where candidates had a good understanding of the language they were using, there was evidence of the use of consistent style throughout including indentation and use of white space.

In general candidates produced code that used meaningful identifiers and appropriate constants and had coded some validation routines. Many candidates included some annotation of their code with more able candidates included annotation that demonstrated their understanding of the problem and solution.

Test strategy

Most candidates were able to describe some type of testing strategy and some evaluation criteria. In future candidates may benefit from considering their objectives when describing their testing strategies and evaluation criteria, ensuring that they plan to test and evaluate against each objective.

Testing

Most candidates were able to design tests that would demonstrate the functioning of parts of their solution. However, some test plans would have benefited from focusing on the logic of the solution rather than repeatedly testing the less complex parts of the system.

In future candidates would benefit from using their objectives and success criteria as a framework for their test plans and ensure that these are met by their solution.

As solutions should be able to carry out particular functions, candidates should ensure that the data entered produces the correct result and the output is correctly formatted. The scenarios provided many opportunities for candidates to test the logic of their solutions using text-based data to produce a mathematical result to their calculations.

All tests from the test strategy should be evidenced with screenshots and discussed rather than stating if a particular test was a pass or fail.

Further development

The specification calls for candidates to:

- consider the outcomes of the testing process in terms of how well the application meetthe objectives set at the beginning of the project
- describe the good features of the application and identifies areas for further development
- provide detailed suggestions for specific extensions to the application.

Many of the discussions produced by the candidates were brief and tended to be narrative rather than reflective and evaluative in nature.

Few candidates offered valid and detailed suggestions for future improvements. However, a minority were able to discuss their solutions in the light of their structure and suggest viable improvements that could be created using their chosen language.

WJEC
245 Western Avenue
Cardiff CF5 2YX
Tel No 029 2026 5000
Fax 029 2057 5994
E-mail: exams@wjec.co.uk
website: www.wjec.co.uk