GCE A LEVEL EXAMINERS' REPORTS

A LEVEL (NEW)
COMPUTER SCIENCE

SUMMER 2019

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

GCE A LEVEL (NEW)

Summer 2019

COMPONENT 1 PROGRAMMING AND SYSTEM DEVELOPMENT

General Comments

Most candidates were well prepared and demonstrated a high standard of knowledge and application required for A2. There was a broad range of answers with many candidates achieving higher marks throughout.

Candidates should be encouraged to expand on their explanations and descriptions when answering knowledge focused questions on topics such as high- and low-level programming languages. Centres should ensure all candidates are able to write, interpret and evaluate specified algorithms in a variety of different formats including flowcharts and pseudocode.

Comments on individual questions/sections

Q.1 The question required candidates to demonstrate their knowledge on low- and high-level programming languages. This question was adequately answered by candidates.

(a) In part (a) many candidates were able to describe how high-level languages are closer to spoken English. Some candidates were then able to describe the ratio of low/high-level code to machine code.

(b) Part (b) was well attempted by candidates and many were awarded a mark for stating a suitable situation where low-level languages could be used. Fewer candidates went on describe this situation fully for example, device drivers - low level languages must be used to directly access memory addresses to fully control hardware.

Q.2 The question required candidates to evaluate an algorithm which searches for duplicate values in an array using Big O notation.

(a) Part (a) was fairly answered by candidates with most being able to successfully determine the growth rate for time performance is O(n^2). Centres should ensure all candidates are able to identify and evaluate the time performance of all algorithms outlined in the specification using Big O notation.

(b) In part (b) most candidates were able to correctly label both axes and draw the gradient for time performance of O(n^2). Some candidates drew an exponential growth rate in place of polynomial. Centres should ensure all candidates are able to illustrate the time performance (time complexity) of algorithms in graphical form.

(c) Part (c) was well answered with many candidates successfully determining the growth rate (space complexity) as constant O(1). This is the final year that O(n) will be condoned, centres should ensure all candidates are able to identify and evaluate the growth rate of all algorithms outlined in the specification using Big O notation.

Q.3 The question requires required candidates to simplify Boolean algebraic expressions through the application of Boolean identities. Both parts of the question were very well answered by candidates. Although, candidates are reminded that they should check their answers methodically. Some candidates correctly applied many Boolean identities although arrived at the incorrect answer due to a single misapplication at an earlier stage.

Q.4 The question required candidates to describe the use of alpha, beta and acceptance testing. This question was very well answered by candidates with most providing accurate and in-depth descriptions.

Q.5 The question required candidates to demonstrate their knowledge of procedures, parameter passing and scope of variables. The question was less familiar to candidates.

(a) In part (a) many candidates were only able to state that procedures were used to avoid duplication of code and are also reusable. Some candidates referred to procedures as functions that do not return a value, this is a distinction that its only apparent in older languages such as Pascal. In more modern high-level programming languages, there is no distinction between procedures and functions, whereby all subroutines are functions/procedures with both terms being interchangeable.

(b) Part (b) was well answered by most with candidates demonstrating a sound knowledge of parameter passing.

(c) & (d)

Parts (c) and (d) were less well answered by candidates. The questions required candidates to analysis the scope and lifetime of variables for a given algorithm. Most candidates were able to describe the purpose of local and global variables, few were able to apply this knowledge to the given scenario. Many candidates defined the scope of TArea as global where the question asked for the lifetime of the variable. Many lost marks for not stating that variable ‘a’ cannot be accessed outside of the subroutine MainProg.

Q.6 The question on proving Boolean expressions using a truth table was very well answered by candidates.

Q.7 The question on XOR encryption was well answered by candidates. Many candidates correctly applied XOR encryption and decryption. Although, some candidates did not fully describe the method used to decrypt the encrypted data by using the same key to retrieve the original data.

Q.8 The question required candidates to produce the a correct BNF definition for a loyalty card code and was well answered by candidates.

Q.9 The question required candidates to write an Insertion Sort algorithm. This algorithm was less familiar to candidates. Many candidates wrote a Bubble Sort algorithm in place of an Insertion Sort which meant candidates were only awarded one to two marks. Centres should ensure all candidates are able to write and interpret all algorithms outlined in the specification in a variety of different formats including flowcharts and pseudocode.

Q.10 The question required candidates to demonstrate their knowledge of verification and validation. The question was mostly well answered by candidates. Most candidates were awarded marks for giving suitable examples of verification and validation. Fewer candidates were awarded marks for correctly defining the terms validation and verification, often providing generic and non-technical definitions e.g. verification/validation is checking if the data entered is correct.

Q.11 The question required candidates to demonstrate their knowledge of translation and execution errors. The question was well answered by candidates and gave a variety of examples including logical errors, syntax errors and semantic errors. Some candidates only stated examples and did not continue to write a fuller definition for runtime and translation errors.

Q.12 Candidates demonstrated a good understanding of data structures.

(a) In part (a) some candidates confused a stack with a queue. Some marks were often lost due candidates not fully describing the operation of a queue and just stating that it uses the FIFO principle. Some candidates did not give technical descriptions of these operations relying more on generic descriptions such as people in a shopping queue.

(b) Part (b) was fairly well answered by candidates. Some candidates were unsure how to represent an array and pointers in graphical form and drew a linked list instead.

(c) Part (c) was well attempted by candidates. Many candidates lost marks as they decremented the front pointer instead of incrementing.

Q.13 Candidates demonstrated a good understand of constructing binary trees. Parts (a) and (c) were very well answered. Part (b) required candidates to represent the binary tree as a two-dimensional array, often candidates presented the data in the correct order but were unfamiliar with representing the left and right pointers in numerical form.

Q.14 This question required candidates to describe and give examples of current legislation and its impact on private data. Few candidates achieved high marks in this question. Most candidates were awarded marks for stating various piece of legislation including the Data Protection Act, The Regulation of Investigatory Powers Act and General Data Protection Regulation. Fewer candidates then went on to discuss the how this legislation impacts on personal data and privacy. Candidates were able to offer suitable examples of measures that can be taken to protect private data. Centres are encouraged to ensure that all topics are explored in enough depth to allow for an enriched discussion on topics with supporting real-world examples.

Summary of key points

Contained within comments on individual questions/section (as above).

COMPUTER SCIENCE

GCE A LEVEL (NEW)

Summer 2019

COMPONENT 2 COMPUTER, ARCHITECUTRE, DATA, COMMUNICATION AND APPLICATIONS

General Comments

Many candidates were well prepared and demonstrated a wide knowledge of the topics in the specification. Good answers were seen for questions requiring precise answers to programming or mathematical problems, and for questions requiring descriptive answers.

In descriptive questions, candidates sometimes provided answers which were correct in general terms but lacked sufficient detail to gain high marks. Where appropriate, credit can be gained by describing specific examples of computing applications.

Comments on individual questions/sections

Q.1 Most candidates showed an understanding of the fetch-execute cycle. Where marks were lost, this was often through focussing only on the processor registers and not discussing the role of the bus system. Few candidates indicated clearly that the Memory Data Register forms an interface between the data bus and the processor, whilst the Memory Address Register forms an interface with the address bus.

Q.2 **(a)** In part (a), a pleasing number of candidates correctly found the nearest floating point representation of the given value, and went on to correctly calculate the absolute and relative errors.

(b) The subtraction task involving two's complement numbers in part (b) was answered well. A minority of candidates attempted to use sign and magnitude representation, rather than two's complement.

Most candidates correctly demonstrated multiplication using an arithmetic shift.

Q.3 The question relating to batch processing and real time transaction processing was answered adequately by most candidates.

Correct examples of batch processing were given, generally involving payroll or utility billing systems. Where marks were lost, this was often through a lack of detailed description. There was a missed opportunity to explain the roles of the master file and transaction file during calculations and updating.

Correct examples of real time transaction processing were given, such as: theatre ticket or hotel room booking systems. A minority of answers incorrectly described real time machine control applications. Candidates often mentioned the need to avoid double bookings. However, few explained in detail how provisional bookings would be made for a limited period of time to allow the customer to enter payment, but would be released again if payment was not received.

Q.4 There was generally some confusion about the operation of switches and routers, although some candidates showed a high level of technical knowledge of these networking devices.

Few answers clearly distinguished between the role of a switch in connecting devices on a local area network by means of MAC addresses, and the role of a router in connecting devices on different networks by means of IP addresses. Relatively few candidates referred to the use of routing tables to allow a router to forward messages through a wide area network by the lowest cost route.

A majority of candidates were aware of the role of a multiplexor in combining signals for simultaneous transmission over a communication link. Some candidates gave good explanations of time division multiplexing and frequency division multiplexing.

Q.5 The assembly language question was generally well answered. A common error was to repeat the input loop only if a positive remainder occurred after subtraction of the correct door entry code. An incorrect door code could equally well produce a negative remainder after subtraction.

Some answers included additional assembly language commands or memory locations not mentioned in the question. Candidates are reminded to use only the commands and memory locations specified.

Q.6 Most candidates produced adequate answers to the questions on robotics.

(a) In part (a), some answers showed a confusion about the definition of robotics. A number of the examples given related to standard factory automation, such as the filling of containers with a specified volume of liquid, which do not involve the use of robots

(b) Candidates were generally aware of the advantages of robotic production for the end users of the products.

(c) In part (c), answers often referred to workers losing their jobs as creating a problem for the employer. However, reducing the workforce and making a saving in wage costs often provides a commercial advantage to a company.

Q.7 Adequate descriptions of high level scheduling of jobs in a multi-programming system were given by many candidates, although a minority of answers showed some confusion with the processing of a sequence of program commands in the fetch-execute cycle, or the handing of interrupts.

Candidates were generally aware of the running, runnable and blocked states of a process. There was some confusion about the role of polling, which involves the operating system checking the status of a process.

Q.8 This question was generally poorly answered. Whilst candidates showed an understanding of the terms 'fixed length record' and 'variable length record', there was often little appreciation of how these different types of record might be used in programs.

(a) In part (a), few candidates explained that the start position of any fixed length record in a file can be calculated by multiplying the record length by the record sequence number.

(b) In part (b), many candidates appreciated that the use of variable length records could save storage space. However, a common error was to suggest that variable length records should be used to store names and addresses which might contain different numbers of characters. Variable length records would only be required if data sizes could vary widely, for example: in storing written reviews of hotels or long descriptions of products.

Q.9 The question on SQL was well answered by many candidates. Where marks were lost, this was often through not following correct SQL syntax in relation to key words or word order.

(a) A particular confusion relates to the use of quote marks. SQL requires single quotes around text values, for example in an INSERT command, but numeric and Boolean values do not require quote marks. Field names, for example after a SELECT command, must not be enclosed in quote marks.

(b) In part (b), it was acceptable to use two nested SELECT commands. Although not in the specification we accepted answers that linked the FOREST_BLOCK and ACTIVITY tables by means of a JOIN command and used a single SELECT structure. There is no requirement for candidates to learn this.

(c) Where a new table is being created, as in part (c), candidates are reminded of the need to declare a primary key field.

Q.10 The relational database question was well answered.

(a) In part (a), entity-relationship diagrams were generally drawn using the correct 'bird's foot' symbols for one-to-many links.

(b) In part (b), designs were presented in a variety of formats. It is recommended that each table design is given in simple text format, rather than as a diagram. The design should specify: the table name, followed by a written list of field names. The identification of primary and foreign key fields is an essential part of the design process. Key fields should be clearly marked by underlining/overlining or other symbols, with an explanatory key provided.

Q.11 Most candidates gave adequate answers to the question about interrupts. Where marks were lost, this was often due to insufficient detail. Rather than 'hardware errors ' or 'software errors', specific examples were required such as: 'printer out of paper' error or 'file not found' error.

Some candidates stated that interrupts indicate that an error has occurred. The majority of interrupts are signals to the processor generated during normal computer operations and do not involve errors, for example: keyboard or mouse input, programs terminating normally, or file transfers to or from a hard disk being completed.

Q.12 Candidates generally gave adequate answers to the question about expert systems.

(a) In part (a), an outline of the main components of an expert system was expected: the knowledge base, inference engine and user interface.

(b) In part (b), many candidates gave suitable reasons for the provision of an expert system in addition to face-to-face interviews with a careers advisor.

Q.13 The question on hotel security was answered adequately by most candidates. There was a general understanding of the requirements of the Data Protection Act as applied to the hotel.

Where lower marks were awarded, this was often due to focussing exclusively on computer software and hardware aspects of security such as: passwords, virus checking, encryption of data, and the use of a firewall. Higher marks were awarded when candidates discussed aspects of management and staff procedures.

Summary of key points

- In descriptive questions, candidates can often gain additional credit for describing specific computing applications.
- Programming and database questions were generally well answered.
- Some very good answers on computer networking were given by candidates who clearly had practical experience of setting up and using the devices described.
- It was disappointing that many candidates failed to appreciate the roles of fixed length records and variable length records in computing applications.

COMPUTER SCIENCE

GCE A LEVEL (NEW)

Summer 2019

COMPONENT 3 PROGRAMMED SOLUTION TO A PROBLEM

General Comments

Administration

Many projects of a good standard were submitted for moderation this summer. Moderators saw some work of an excellent standard. Many centres had assessed the work accurately and had clearly explained their assessment decisions which aided the moderation process.

This specification requires work to be uploaded. In addition, candidates' functional solutions should also be included in the coursework submission. Centres should ensure that candidates' solutions are presented in a format that allows moderators to run the candidates' programs with ease. In too many instances, moderators found that solutions included absolute rather than relative pathways to files that prevented the solutions from functioning correctly.

It would aid the moderation process if centres would ensure that candidates' work and documentation are saved with filenames that clearly identify the centre number, candidate number and candidate name. As detailed in the specification for this qualification, "For example Diane Smith (centre number 69999, candidate number 12345) would store her work in a folder named 69999_12345_SM_D. In addition, candidates should ensure that they have linked their work to the GCE Computer Science Component 3 Task sheet (U3e).

It is an essential requirement that all candidate work is authenticated, and the authentication sheets are uploaded with the candidates' work. In the recent series much time was spent by moderators and WJEC officers contacting centres to request the missing paperwork.

The Joint Council for Qualifications (JCQ) document "Instructions for conducting non-examination assessments" states that:

*"All candidates **must** sign a declaration to confirm that the work they submit for final assessment is their own unaided work.*

*Teachers **must** sign a declaration of authentication after the work has been completed confirming that:*

- *the work is solely that of the candidate concerned;*
- *the work was completed under the required conditions;*
- *signed candidate declarations are kept on file."*

It should also be noted that any additional candidates' work and/or paperwork requested by moderators should be provided in a timely manner.

Candidates' work

The following information is provided to help centres guide candidates through the NEA in future. There was evidence of some confusion regarding the following sections of the project work.

In general, many centres do not appear to have recognised the importance of the discussion section for the identification of suitably substantive problem situations. This is a good opportunity for teachers to steer candidates away from unsuitable ideas that will lack the scope required to produce work to a standard and level appropriate for this qualification.

During design work candidates should identify the objectives for their problem solutions. These objectives should inform all sections of the candidates' work from this point onwards.

For each objective, candidates should:

- Design input and output facilities and appropriate data structures
- Produce algorithms for processing
- Develop a prototype if relevant and redesign if necessary
- Fully develop the solution
- Testing should cover each objective
- Evaluation of the solution for each objective

The prototype section of work is intended to allow candidates to trial part of their design and to reflect on the method of solution chosen. In many cases, candidates included feedback from others in this section of work. This was not appropriate as this section of the work relies on self-reflection.

The refinement of design section of the work considers third party feedback in addition to self-reflection to move the project forward.

It is essential that feedback in the discussion work and in the refinement of the design work is provided by informed third parties who are able to move the project forward rather than end users.

In some cases, candidates produced final solutions that were over reliant on application packages such as relational databases and spreadsheets. This qualification does not allow the use of such applications other than as a vehicle for storage of files. Candidates should not make use of any of the facilities built into the application and all validation of data and sorts/searches of sets of data must be implemented through the creation of original code.

It is strongly recommended that the use of such application packages is always avoided with candidates developing their own file handling routines and facilities.

Centres should ensure that where candidates' solutions require the use of usernames and passwords that this information is included on the candidates' mark sheets or in a 'readme' document stored with the functional solution. It is essential that moderators are able to run the candidates' solutions to fairly assess the appropriate marks for the work.

Comments on individual questions/sections

Most candidates had chosen suitable problem situations as a basis for their project work. These problem situations would provide them with enough scope to produce a fully working system at an appropriate level for this qualification although this was not always fully exploited.

However, a minority of candidates had chosen problem situations that did not provide the opportunities for data handling that are required to access marks for design, implementation and testing at a level appropriate for this qualification. **It is not appropriate for candidates to undertake problem situations that involve the creation of games or quizzes.**

The specification has been designed to provide two opportunities for feedback from teachers, competent third parties and peers that should have encouraged these candidates to refine or change their choice of problem as they will not be able to access the full range of marks.

Candidates should consider whether their choice of problem situation provides them with sufficient:

- Opportunities to carry out an investigation in appropriate depth to provide evidence to allow them to complete the analysis, problem definition and objectives sections of the work to an appropriate level of complexity for an A2 qualification.
- Complexity to provide the opportunities needed to access the full range of marks
- Data handling process to allow thorough testing processes to take place

Discussion

It is important that centres recognise the importance of the discussion section. This section provides opportunities for the candidates to present their problem situations to their teacher, peers and/or other competent third parties. Candidates should receive detailed informed feedback regarding the scope of their chosen problem and should reflect, in depth, on the discussions and feedback to allow them to firm up their ideas and ensure that unsuitable topics are revised or discarded. In many cases candidates report feedback but don't include their reflections on the feedback or a justification of their decisions to accept or reject specific feedback.

The preparation of the materials for the presentation/discussion provides opportunities for the candidates to reflect on their ideas and the problem situation. If necessary, the candidates can reframe their problem situation or even identify a different, more appropriate problem situation.

Investigation

Where candidates had chosen suitable real-life problems, they have the opportunity to carry out an investigation into the current system. Candidates should identify the data collected, processed and output by the current system. In many cases candidates did not carry out this investigation and analysis but provided narrative accounts of problem situations that did not allow them to identify suitable objectives for their solutions nor form the basis for a comprehensive design.

All candidates are required to carry out desk-based research into similar commercial solutions created to solve similar problems. In many instances, candidates are paying only lip service to this requirement.

This research is an important part of the project as it should inform the design process. In addition, it is essential to note that the final section of the NEA requires candidates to evaluate their final solutions against the commercial systems. This is intended to provide them with informed ideas for further development of their systems.

Where candidates are not able to identify a real-life problem, they should carry out extended research into similar commercial systems, identifying common characteristics and should base the conclusions of their investigations on the information that they have been able to gather.

It is important that candidates produce a comprehensive working specification and that measurable objectives are set that will inform the design, prototyping and testing processes.

Prototype

Candidates should identify the areas to be prototyped. These areas should cover the essential sections of the solution.

Candidates should not include features such as logon facilities and validation that will complicate the prototype development and testing process. It is not necessary to include all fields for data files. Centres should note that the extent of the prototype will reflect the nature of the chosen problem.

The prototype work is intended to allow self-reflection on the chosen method of solution and the design work. It is not appropriate to include feedback from third parties in this section of the work.

Post-prototype refinement of design

This part of the work is intended to allow candidates to consider third party feedback and to decide what changes, if any, should be made to the original design. This feedback should come from competent third parties and not from end users who are likely to lack the technical knowledge to give the constructive advice required to refine the work to date. Candidates should justify their acceptance or rejection of feedback.

Testing

It is important that the testing work should focus on the functionality of the solution in terms of:

- Input facilities including measures to ensure reasonable data entry
- Processing facilities to ensure correct and accurate output
- Appropriate output including screen and paper-based outputs

The testing work should cover each objective with data designed to measure the outcomes of the system against the desired outcome. The quality of the commentaries accompanying the testing evidence has a major role in identifying the marks to be awarded for this section of the work.

Evaluation

The evaluation section should cover the effectiveness of the programming language and a justification of the tools and techniques used.

Candidates should then compare and contrast their completed solutions with the commercial systems considered during the investigation section of the work. This comparison should allow candidates to identify and discuss the good features and shortcomings of their work. It is important that candidates describe significant potential improvements to their systems that would more reflect the facilities of the commercial solutions to the chosen problem.

Candidates should also consider their own strengths and weakness and how they would adapt their approach to improve their performance if faced with a similar task in the future.

Summary of key points

Contained within comments on individual questions/section (as above).

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