GCE EXAMINERS' REPORTS

GCE (NEW)
COMPUTER SCIENCE
AS/Advanced

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

General Certificate of Education (New)

Summer 2019

Advanced Subsidiary/Advanced

UNIT 1 FUNDAMENTALS OF COMPUTER SCIENCE

General Comments

This is the fourth year of awarding the new reformed specification.

Candidates found Unit 1 demanding again this series and a mean mark below 40 confirms this. This mean is however in-line with previous series. Relatively few candidates achieved a mark over 75.

At question level, it was pleasing to note that candidates performed well in some questions and there is a high percentage of attempts for each question. However, ten of the thirteen questions have a facility factor below 50.

Comments on individual questions/sections

Q.1 Around half of candidates were able to define the term Internet and describe the use of the given networking protocols.

Q.2 Only a few candidates were able to state the correct outputs that the algorithm will give, with many incorrectly providing an output for c and d. A lack of terminology was seen when candidates attempted to explain why the algorithm does not work as intended. Very few candidates used terminology such as scope and lifetime.

Q.3 It was very pleasing to see over three quarters of candidates describe the function of three main components in a contemporary Von Neumann type CPU architecture. Many candidates also went into depth by describing the function of different register such as MAR and MDR, which goes beyond the requirements of the specification for this unit, but should stand them in good stead for A2 units.

Q.4 Around a third of candidates described the functional characteristics of a HDD well. Some candidates only described more general characteristics, such as durability and speed of access. This was condoned this series, but may not be in future series.

Over half of candidates were able to explain why there would be a difference in disk access speeds when loading File A and File B into main memory. These were also able to explain why a Solid State Drive (SSD) does not have the same issue as a HDD.

Q.5 Many candidates were able to give the simplest Boolean expression for the first two truth tables. Very few were able to give the simplest expression for Q5c.

Q.6 This was the second worst answered question of the paper with only a fifth of candidates able to answer question about parameter passing. For many centres, candidates seemed completely unfamiliar with the term. A significant number of candidates failed to attempt the question.

Q.7 This was the worst answered question of the paper. Only a very few candidates were able to answer each part of the question with over a tenth of candidates not attempting the question at all. Very few were able to draw a diagram that shows how a transaction file and master file are used during an update, with only a few illustrating the correct outputs.

Candidates were unable to apply their knowledge and understanding to describe the data used an application of their choice and were also unable to describe the most suitable mode of operation for your chosen application. Very few candidates could name a mode of operation.

Q.8 It was pleasing to see over half of candidates simplify the given Boolean expression using Boolean algebra and identities. Where candidates encountered problems was with the A.(0 + $\bar{A}$) aspect of the expression, incorrectly dealing with the 0.

Q.9 Less than half of candidates were able to describe methods used in file security to prevent accidental data loss from computer systems. Many gave nothing more than a generic response relating to backups.

Q.10 It was very pleasing to note that almost two-thirds of candidates had good knowledge of both binary and linear search algorithms. They were able to dry-run the given algorithm correctly and then write their own in-line with the example given. These candidates were also able to describe an appropriate circumstance for the use of each search type.

Q.11 Candidates performed worse in this mathematical content question than in previous series, which is a shame as this has tended to be well answered. Only a few of marks on average were awarded to candidates for this question, with a significant number of candidates being awarded 0 marks for Q11b. Many candidates seemed unfamiliar with the term range. Candidates performed well in Q11a, but could only give one advantage in general of representing numbers in integer form and in floating-point form in Q11c.

Q.12 Candidates tended to either perform really well when describing the different methods of investigation used by a systems analyst or really poorly and this tended to be relevant to the whole cohort entered for each centre.

Q.13 In general, candidates gave a written response that had an adequate line of reasoning with elements of coherence, relevance, and logical structure. They showed an adequate understanding of the requirements of the question and a satisfactory knowledge as specified in the indicative content of the mark scheme. These candidates presented a discussion with limited examples and used appropriate technical terminology referring to the indicative content.

Summary of key points

- Candidates demonstrated a good understanding of the main components in a contemporary CPU, Boolean algebra expression simplification and search algorithms. They performed best in these questions.

- Candidates found the questions on parameter passing and file update difficult. This was reflected in the low facility factor for these questions.

COMPUTER SCIENCE

General Certificate of Education (New)

Summer 2019

Advanced Subsidiary/Advanced

UNIT 2 PRACTICAL PROGRAMMING TO SOLVE PROBLEMS

General Comments

Unit 2 is a practical examination with candidates required to demonstrate the application of knowledge and understanding at all times. It was pleasing to see the mean mark for the paper increase and more candidates demonstrate a better understanding of the practical applied programming aspect of computer science.

Comments on individual questions/sections

Q.1 This was well answered by most candidates; however, some candidates could not identify key fields.

Q.2 Candidates lack of understanding of data validation and foreign keys resulted in some candidates losing marks.

Q.3 This question required candidates to justify their choice of software to implement a solution for the company in the scenario i.e. KimsKarate candidates use of technical terminology let them down on some occasions.

Q.4 This flowchart question was assessed differently to previous years and was considered more demanding; however, many candidates have performed well here contrary to expectations.

Q.5: This question on OOP was also assessed for the first time and considered demanding.

Part B:

1. This question was made more demanding by expanding the depth of annotation and understanding required by candidates.

2. This question was also considered to be quite demanding as it had file handling and number handling combined.

Summary of key points

Most of the candidates demonstrated a good understanding of the specification. Questions attempted data was unavailable this series. Many candidates were well or very well prepared for programming and many excellent answers were evident. There was also evidence that some candidates had been well prepared for the majority of the practical programming elements found in section B, however, performance when using file handling code was again disappointing, and performance whilst annotating code was variable

COMPUTER SCIENCE

General Certificate of Education (New)

Summer 2019

Advanced Subsidiary/Advanced

UNIT 3 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 wide range of answers with many candidates achieving higher marks in questions exploring data structures and depicting truth tables. Candidates demonstrated significant improvement in their responses than in previous years.

Candidates should be encouraged to include more examples when answering knowledge focused questions such as the question on natural language interfaces. Centres should also encourage a deeper understanding of topics such as programming paradigms and translators.

Comments on individual questions/sections

Q.1 Candidates demonstrated a good understanding of queues, stacks and linked list data structures. The question was very well answered by candidates. In part (a) some candidates confused a stack with a queue and vice versa. Some marks were often lost due to candidates not fully describing the operation of the data structures, just stating the FILO and FIFO principles. Some candidates did not give technical descriptions of these operations relying more on generic descriptions such as a stack of plates or a supermarket queue. Part (b) was very well answered by most candidates.

Q.2 The questions on Boolean algebra were well answered, many candidates simplified their expressions to the simplest form using a variety of methods. Candidates are reminded that they should check their answers methodically. Some candidates correctly applied many Boolean algebraic identities although arrived at the incorrect answer due to a single misapplication at an earlier stage.

Q.3 The question on masking was well answered by candidates. Many candidates correctly applied an AND or XOR mask. Although, most candidates did not successfully determine the most significant bit as being the left most bit but instead identified the right most bit.

Q.4 **(a)** The question on natural language interfaces was well answered in parts by candidates. In part (a), many candidates identified natural languages interfaces as the using of speech and linguistics, but many did not clearly that natural language interfaces use speech and linguistics to interact and control a software application.

(b) In part (b) many candidate offered a suitable example include handsfree car systems and smart speakers.

However, fewer candidate went on to fully describe how this technology is suited to a natural language interface.
Some candidates often confused natural language interfaces exclusively as language translation software.

(c) Part (c) was very well answered by most candidates and often described more than three potential problems with natural language interfaces. However, some candidates did not respond well to this question well due to the lack of understanding of the term natural language interface.

Q.5 The question proved demanding for many candidates as it required them to follow an algorithm and fully describe its purpose, elegance and characteristics. Most candidates identified the algorithm as a recursive Quick Sort.

(a) Part (a) was well answered with most candidates stating the algorithm sorts an array, fewer candidates then went on to state it sort it in ascending order. There were some candidates that stated the algorithm identifies high and low temperatures due to the use of the temp variables tmpSwap, tmpLow and tmpHi, centres should ensure that candidates quickly dry run the algorithm to identify its purpose and not look for meaningful identifiers to speculate its purpose.

(b) Part (b) was very well answered by those candidates who identified the algorithm as recursive or as a Quick Sort.

(c) Part (c) was less well answered by candidates with many misinterpreting the term elegance in relation to recursive algorithms. Most candidates were awarded a mark for identifying that it solves the problem in a shorter time span than a non-recursive algorithm. Fewer went on to discuss the algorithm's decomposition of a larger problem and compactness of the solution.

Q.6 The question on shortest path algorithms and traversal costs were less familiar to candidates.

(a) In part (a) most candidates were able to describe the shortest path algorithm as being the shortest path between to vertices or nodes. Fewer went on to describe that each path has a weighted cost and the path with the lowest total cost is the shortest path.

(b) In part (b) many candidates answered the questions although few managed to accurately depict every correct path and every correct cost. Those that were familiar with least cost analysis answered the question very well. Many candidates understood the least cost analysis but were unable to demonstrate their understanding in written form.

(c) Part (c) was well attempted by candidates but many candidates forgot to include the cost of traversing the actual node (+2). Therefore many responses identified the shortest path as {a,c},{c,d},{d,e} although miscalculated the total cost. Centres should be reminded to ensure that candidates fully read the question as they will vary each series.

Q.7 The question on programming paradigms was adequately answered by candidates.

(a) In part (a) some candidates were able to fully describe the term programming paradigm. Many were able to state a programming paradigm was a type of programming languages, less were able to go on to describe the need to different paradigms.

(b) Part (b) was well answered by many candidates with most clearly describing the difference between the given paradigms and included suitable descriptions of their purpose.

(c) Part (c) was well answered by most candidates with many being able to offer suitable examples of programming paradigms many included answers such as event driven programming and object orientated programming. Only some candidates were able to go on to provide a suitable software application where each paradigm is appropriate. Centres should ensure that candidates fully understand the purpose and need for different programming paradigms in more depth.

Q.8 The question on proving Boolean expressions using true table was very well answered by candidates.

Q.9 The question required candidates to evaluate an algorithm using Big O notation.

(a) Part (a) was less familiar to candidates with many unable to evaluate a logarithmic algorithm and many offering answers similar to those in previous papers e.g. bubble and insertion sorts. Those candidates who were able to identify the algorithm as a logarithmic binary search, described and evaluated it very well. Centres should ensure all candidates are able to identify and evaluate all algorithms outlined in the specification using Big O notation.

(b) In part (b) most candidates were able to correctly draw both axis and the time performance of O(n). Centres should ensure all candidates are able to illustrate the time and space performance of algorithms in graphical form.

(c) Part (c) was well answered by candidates with the many identifying Search B as the most efficient algorithm.

Q.10 The question on compression was well answered by the majority of candidates. Candidates responses included descriptions of lossy and lossless compression. Few candidates went on to fully describe compression ratios.

Q.11 This question was focused on Object-Orientated Programming and its features.

(a) Part (a) was well answered with candidates describing classes and objects and the relationship between them. Fewer candidates described the use of properties and instance variables.

(b) Part (b) was less well answered by candidates. Candidates were required to explain the relationship between classes and instances. Most candidates were able identify that a class in a template of an object and instance is variable holding the memory address and copy of an object.

(c) In part (c) candidates demonstrated a very good understanding of the term method. Fewer candidates describe a method only being able to access its own objects, data and encapsulation.

Q.12 The question required candidates to follow a flowchart and represent it in pseudocode, identifying any mistakes. Many candidates were awarded marks for correctly converting the flowchart into pseudocode. Fewer candidates then went on to correctly place and correct the code to receive higher marks. Centres should ensure all candidates are able to write and interpret algorithms in a variety of different formats including flowcharts and pseudocode.

Q.13 This question required candidates to describe and give examples of different translators and distinguish between them. Few candidates achieved high marks in this question. Most candidates were awarded marks for outlining the basic features and purpose of compilers, interpreters and assemblers. Fewer candidates then went on to discuss the varying distinguishing features of each and offer examples of their use. 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

General Certificate of Education (New)

Summer 2019

Advanced Subsidiary/Advanced

UNIT 4 COMPUTER ARCHITECTURE, 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 **(a)** In part (a), the role of buffering was understood in general terms, but few candidates clearly stated that a buffer stores data temporarily during input or output operations to allow for differences in operating speed between a fast processor and slower peripherals.

(b) Most candidates gave adequate answers to the question about interrupts in part (b). 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.2 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.3 The question on SQL was well answered by a number of candidates. Where marks were lost, this was often through not following correct SQL syntax in relation to key words or word order.

In part (d), it was acceptable to use two nested SELECT commands, we also accepted linking the COURSE and MODULE tables by means of a JOIN command and use a single SELECT structure although there is no requirement for candidates to learn the JOIN command.

Q.4 The question on binary numbers was well answered, with many candidates correctly converting base-10 numbers to floating point and two's complement integer formats.

Absolute errors were found correctly by a majority of candidates and good descriptions were given of relative accuracies as a result of rounding or truncation, before or after multiplication.

Q.5 The assembly language question was generally well answered. A common error was to omit the initialisation of ***total*** and ***count*** variables before input of data begins.

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 The indexed sequential file question caused difficulties for a number of candidates.

In part (a), a common mistake was to assume that all necessary index blocks were already present. The question indicated that index blocks may need to be created before new records could be stored.

A number of candidates assumed that the entire data file was sorted into order of key fields, and a sequential update of the whole file was necessary to insert the additional records. In practice, each data block would be ordered sequentially, but data blocks would be added as necessary and would not be stored in any particular order.

Q.7 The calculation involving fixed and variable length records was carried out accurately by many candidates.

Q.8 The question on data mining was answered adequately by many candidates.

Data mining is defined as the analysis of large amounts of data, perhaps originally collected for other purposes, in order to identify patterns which are useful to guide decision making. This would include for example: the analysis of shop sales to identify the most suitable products to stock, the analysis of patterns of crime to identify areas where policing should be focussed, or the analysis of road accident statistics to determine appropriate insurance premiums for different categories of driver.

There was no intention that the results of data mining would identify specific persons. A number of candidates incorrectly stated that data mining allowed advertising to be targeted at individual customers based on analysis of their previous purchases.

Q.9 The question on data validation methods was well answered by most candidates. Occasional answers described verification, such as double entry of data items, rather than validation.

Q.10 The question on malicious software was generally well answered. Candidates showed a knowledge of the different categories of malicious software, how these were introduced to computer systems, and appropriate methods of protection.

Q.11 **(a)** The question relating to the automated railway system in part (a) was adequately answered by many candidates, but answers often lacked detail. Few candidates described feedback loops, for example: a sensor monitoring the train speed, then accelerating or braking to maintain the speed within required limits; a sensor monitoring the current location of the train, and reducing speed on approach to a station.

(b) In part (b), candidates generally outlined few of the important principles for safety critical systems. A full answer might mention: thorough testing of the system before introduction; redundancy in the event of software or hardware failure; fail-safe procedures, which might include handing over manual control in the event of system failure; regular maintenance and testing; and security to prevent unauthorised access to the computer system.

Q.12 Few candidates gave detailed descriptions of the operation of a mainframe computer which included references to scheduling, time-slicing and the running, runnable and blocked states of processes.

In some answers there appeared to be confusion with a multi-programming personal computer, rather than a central computer linked to a series of user terminals.

Q.13 Descriptions of distributed processing often gave a general definition but lacked detail. Examples given by candidates often related to scientific programs which could be downloaded and run as screen savers on personal computers. However, little detail was given of the objectives of the research projects, the calculations being carried out by individual machines, or the overall results obtained.

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.

COMPUTER SCIENCE

General Certificate of Education (New)

Summer 2019

Advanced Subsidiary/Advanced

UNIT 5 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 68999, candidate number 12345) would store her work in a folder named 68999_12345_SM_D. In addition, candidates should ensure that they have linked their work to the GCE Computer Science Unit 5 Task sheet (U5e).

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 dot 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