



---

# **GCE A LEVEL EXAMINERS' REPORTS**

---

## **COMPUTER SCIENCE A LEVEL**

**SUMMER 2018**

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.

| <b>Unit</b> | <b>Page</b> |
|-------------|-------------|
| Component 1 | 1           |
| Component 2 | 4           |
| Component 3 | 7           |

## **COMPUTER SCIENCE**

### **GCE A LEVEL**

**Summer 2018**

#### **COMPONENT 1: PROGRAMMING AND SYSTEM DEVELOPMENT**

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 such as the question on the use of procedural programming languages.

#### **Individual questions**

##### **Question 1**

The question required candidates to complete a truth table. This question was very answered by candidates.

As a side note, some candidates provided answers on the actual examination paper and candidates in future should be reminded that all answers must be written in the answer booklet provided or they may not be accepted.

##### **Question 2**

The question required candidates to demonstrate their knowledge on code optimisation. In part (a) although most candidates described what is meant by the term code optimisation only a few went on to describe how code optimisation works by replacing high-level programming language constructs with efficient low-level programming codes. Many candidates were able to identify the objective of code optimisation, with many referring to the decreasing demand on resources and increasing the overall speed of the program.

In part (b) many candidates were able to identify why the block of code had not been optimised as it repeatedly assigns to the variable 'item'. Fewer candidates then went on to explain how to optimise the code.

##### **Question 3**

The question required candidates to demonstrate their knowledge on feasibility studies and investigations into computer systems. Overall the question was answered well with many candidates explaining that a feasibility study is used to advise on numerous factors including technical practicability and cost effectiveness. Part (b) was also well answered but even though candidates identified observation and documentation they failed to identify how they can be used as a method of investigation e.g. inspection of business documents and user manuals.

#### Question 4

Part (a) required candidates to equate Boolean identities. This was very well answered by candidates.

Part (b) required the simplification of Boolean expression and was well answered, many candidates simplified their expressions to their simplest form using a variety of methods.

#### Question 5

The question required candidates to describe the advantages of procedural languages to write a program based on an algorithm. Many candidates responded to this question by successfully describing the characteristics of procedural programming languages although fewer candidates went on to describe the advantages when using it to implement a program based on an algorithm. Some of the responses that candidates could have given include how procedural languages provide closer control of the underlying operation of that hardware and that they also enable similar operations to be repeatedly called at varying stages of execution.

#### Question 6

The question required candidates to demonstrate and apply their knowledge of shortest path algorithms, calculate the routing costs of networks and identify the shortest path between two nodes. Part (a) was well answered with candidates providing accurate explanations of the term shortest path algorithm although many failed to identify that it uses a weighted network to produce costs.

Part (b) was well answered with candidates accurately drawing a network from the given table. Most candidates then went on to identify the shortest path between the two given nodes successfully.

#### Question 7

The question required candidates to demonstrate their knowledge on the stack data structures referring to pop and push functions and a stack underflow/overflow. Part (a) was a very well answered question with candidates clearly describing a stack. Parts (b) and (c) were less well answered with candidates not clearly defining a pointer to be incremented and decremented in their algorithms and very rarely providing suitable output messages if a stack was empty or full.

#### Question 8

This question required candidates to produce the a correct BNF definition for a product code and was well answered by candidates.

#### Question 9

The question required candidates to demonstrate their knowledge of programming constructs. This was a well answered question and most candidates described a selection construct although many did not describe the effect of using one in that a program can take different paths based on different conditions.

### Question 10

The algorithm question required candidates to write an algorithm that calculated the NI contributions of employees. This was a well answered question with many candidates providing suitable algorithms. Some candidates provided suitable solutions without declaring variables that had suitable data types. There were many candidates who provided accurate sorting through the implementation of a bubble sort within their algorithms.

### Question 11

The question required candidates to demonstrate and apply their knowledge of the binary tree data structure. Candidates demonstrated a very good understanding of constructing binary trees and applying a post-order traversal.

### Question 12

The question required candidates to demonstrate and apply their knowledge of evaluating the efficiency of algorithms and the use of Big O notation. Part (a) was well answered with many candidates successfully identify the order of efficiency as constant  $O(1)$  and providing a suitable example such as accessing the first item in array.

Part (b) was very well answered by candidates and many were awarded marks for correctly labelling the axes.

Part (c) Most candidates provided suitable examples of a  $O(N^2)$  efficient algorithm either through explaining the use of nested loops or identifying an appropriate algorithm such as a bubble sort.

Part (d) was well answered by candidates when identifying the efficiency as logarithmic although fewer candidates explained what is meant by a logarithmic relationship between time and size of data, although many did provide a suitable example such as a binary search.

### Question 13

The question required candidates to construct an identifier table for a given segment of code. The question was well answered by candidates although in part (a) some forgot to declare appropriate data types in the user identifier/variable table. Part (b) was again very well answered and candidates fully understood what was required to translate a statement of code into hexadecimal tokens.

### Question 14

There was a wide variety of responses to this question. The question required candidates to describe the main stages in program production **between** investigation and acceptance testing. Many candidates successfully drew on their knowledge from across their entire computer science course and in particular their non-examination assessment. Although most candidates provided detailed responses a large number included the investigation and acceptance testing stages for which no marks could be awarded.

## COMPUTER SCIENCE

### GCE A LEVEL

Summer 2018

#### COMPONENT 2: COMPUTER ARCHITECTURE, DATA, COMMUNICATION AND APPLICATIONS

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 are encouraged to go beyond a theoretical description of computing techniques by referring to practical examples from commerce or industry, and to include diagrams where appropriate. A good example can often lead to a candidate gaining marks they wouldn't otherwise attain.

#### Individual questions

- Q1. 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 (b), it was acceptable to use two nested SELECT commands, or to link the 'voyage' and 'container' tables by means of a JOIN command and use a single SELECT structure.

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

- Q2. Candidates generally appreciated the events which should take place to animate the character, but few described these sufficiently precisely in computing terms.

A series of shift operations must be applied to the set of bytes representing the character, separated by short time intervals, in order to create the animation. In part (a), logical left shifts are applied so that the empty **least significant bit** positions are filled by binary 0 values.

In part (b), cyclic right shifts are applied so that the bits overflowing from the **least significant bit** positions are automatically replaced in the empty **most significant bit** positions.

- Q3. In part (a), many candidates identified the multiple occurrence of data items in the tables, but relatively few identified ApplicantName and InspectorName as the fields causing redundancy.

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

In part (b)(ii), designs were presented in a variety of formats. It is recommended that each table design is given as: the table name, followed by a written list of field names. Candidates are reminded that 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.

- Q4. A majority of candidates correctly identified batch processing as the most suitable mode of operation for gas billing. However, few gave a detailed description of the production of gas bills by: loading meter readings from a transaction file, calculating the quantity and cost of gas used, then adding customer name and address data from a master file.

Good answers were given for traffic light real-time control systems, and theatre booking real-time transaction processing.

- Q5. Candidates were generally familiar with Internet protocols.

In part (a), some answers indicated confusion about the roles of IMAP and SMTP in e-mail systems. SMTP allows e-mail to be sent from a client computer to an Internet server, and for sending and receiving e-mail between servers. IMAP (Internet Message Access Protocol) allows the retrieval of copies of e-mail messages from the destination server to a client computer.

Many candidates correctly identified TCP as the most suitable protocol for FTP file transfer and UDP for video streaming, explaining the choice in terms of the presence or absence of error checking and re-transmission of missing data packets.

A number of good answers were seen in part (d), explaining the role of DHCP in allocating IP addresses to devices as they connect to a network, then later releasing the addresses for re-use.

- Q6. In part (a), a number of candidates commented that a single-processor computer would take too long to run the meteorological model, but did not give the essential reason: a slow model may not produce its results until after a forecast weather event had already occurred.

In part (b), some candidates applied Amdahl's formula whilst others solved the problem from first principles, arguing that: 20% of the processing time (1 hour) involves linear processing, whilst the remaining time (4 hours) could be shared simultaneously between 12 processors, adding a further 20 minutes of processing time. It was noticeable that the candidates arguing from first principles generally gave correct answers, whilst many candidates using a formula produced obviously incorrect answers ranging from a few seconds to several days. It is therefore recommended that teachers encourage a full understanding of the principles of linear and parallel processing, rather than simple memorisation of a formula.

- Q7. The assembly language programming question was well answered by a number of candidates. Teachers are to be commended on an improvement in the ability of candidates to handle jump operations.

Where marks were lost, this was often due to errors during the calculation steps. Candidates are advised to carefully read the descriptions of the assembly language commands and note the destination of the data. For example: MOV A,B would copy a value **from** register A **to** register B; ADD A,B adds values in registers A and B, **leaving the result** in register A. Few candidates provided a data table listing the variables in the original algorithm and the corresponding registers which will hold these values. Use of a data table can be helpful in checking an assembly language program and identifying logical errors.

Q8. The questions on floating point and two's complement binary numbers were correctly answered by a majority of candidates.

Q9. In part (a) most candidates gave valid examples of interrupts, although some answers lacked detail. Rather than 'hardware errors' or 'software errors', specific examples were required such as: 'printer out of paper' error or 'file not found' error.

In part (b), many candidates were uncertain about the role of polling in a spooling system. Polling may be considered as the opposite of an interrupt, with the operating system directly checking the state of processes rather than waiting to receive notifications. This might involve regularly checking the availability of output devices, and determining whether jobs had completed data transfer and were ready to re-join the runnable job queue.

Candidates were often imprecise in explaining the role of buffers. A buffer allows data transfer between devices with different processing speeds, so that the faster device can work on other tasks whilst the slower device is completing its data transfer.

Q10. Random access and indexed sequential files were explained well by many candidates. However, a significant number of candidates incorrectly described the terms as referring to hierarchical storage of files on the hard drive of a personal computer.

In part(b), indexed sequential files were often described by candidates as storing all records in order of key field, with re-sorting of the complete file necessary each time that a record was added or deleted. However, this process would be very inefficient. Indexed sequential files typically store records in key field order only within particular data blocks. Access to the appropriate data block is provided by the index system. Data blocks are added as necessary using available memory space, and need not be stored in sequential order.

Relatively few candidates produced diagrams to illustrate multi-level indexed file systems. Diagrams would make many answers clearer, and would gain credit.

Q11. Candidates generally showed a good understanding of the components of contemporary personal computer systems, and demonstrated detailed practical knowledge of storage devices, electronic memory, and peripheral interfaces.

Q12. Most candidates had a good understanding of the term 'expert system', though a significant number incorrectly interpreted this as referring to any complex software for undertaking a specialist task, such as: business accounting or scientific modelling. A majority of candidates chose medical diagnosis as an example of an expert system. Advantages of the system were clearly stated. However, descriptions of the use of system were often insufficiently detailed. It is recommended that candidates make a detailed case study of an expert system, so that they are able to describe the user interface, the sequence of interactions with the user, and the output of results and advice from the system. Features of the expert system which might be discussed include: the ability of the system to explain its reasoning; the presentation of different possible conclusions, with the relative probabilities of each being correct; and an ability to receive feedback from the user so that learning can take place to improve future decision making.

## COMPUTER SCIENCE

### GCE A LEVEL

Summer 2018

#### COMPONENT 3: PROGRAMMED SOLUTION TO A PROBLEM

##### General Comments

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.

This specification requires work to be submitted electronically. In addition, the 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 is an essential requirement that all candidate work is authenticated. 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."*

When uploading candidates' work for moderation centres must ensure that the completed mark sheets and authentication sheets are provided for each candidate. In the recent series much time was spent by moderators and Eduqas officials contacting centres to request the missing paperwork.

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 had not recognised the importance of the discussion section to 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.

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 in particular MS Access. This qualification does not allow the use of applications other than as a vehicle for storage of files. Candidates should not make use of any of the facilities built into an application and all validation of data and sorts/searches of sets of data must be implemented through the use of original code.

### **Component 3 – Non Examination Assessment**

Most candidates had chosen suitable problem situations as a basis for their project work. These problem situations provided them with sufficient 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 chose problem situations that did not provide the opportunities for data handling that are required to access marks for design, implementation and testing at this level. It is not, for example, appropriate for candidates to develop on-screen games devoid of data handling facilities. The specification has been designed to provide two opportunities for feedback from teachers and peers that should have encouraged these candidates to refine or change their choice of problem.

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 processes to allow thorough testing procedures 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 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.

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

### **Investigation**

Where candidates had chosen suitable real-life problems they had the opportunity to carry out an investigation of 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.

This specification requires all candidates to carry out desk-based research into similar commercial solutions created to solve similar problems. This is an important part of the project as it should inform the design process. In addition, candidates are required to evaluate their final solutions against the commercial 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 to inform the design, prototyping and testing processes.

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
- develop testing that covers each objective
- create an evaluation of the solution for each objective

## **Prototype**

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

Candidates should not include facilities 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.

## **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. It is important that candidates realise that this section of work requires refinement of design not redesign.

Candidates should justify their acceptance or rejection of feedback.

## **Testing**

It is important that the testing work focuses 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 actual outcomes of the system against the desired outcomes. The quality of the commentaries accompanying the testing evidence have 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.



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