

GCE A Level Examiners' Report

Computer Science
A Level
Summer 2025

© WJEC CBAC Ltd. 2025



Introduction

Our Principal examiners' report provides valuable feedback on the recent assessment series. It has been written by our Principal Examiners and Principal Moderators after the completion of marking and moderation, and details how candidates have performed in each component.

This report opens with a summary of candidates' performance, including the assessment objectives/skills/topics/themes being tested, and highlights the characteristics of successful performance and where performance could be improved. It then looks in detail at each unit, pinpointing aspects that proved challenging to some candidates and suggesting some reasons as to why that might be.¹

The information found in this report provides valuable insight for practitioners to support their teaching and learning activity. We would also encourage practitioners to share this document – in its entirety or in part – with their learners to help with exam preparation, to understand how to avoid pitfalls and to add to their revision toolbox.

Further support

Document	Description	Link
Professional Learning / CPD	Eduqas offers an extensive programme of online and face-to-face Professional Learning events. Access interactive feedback, review example candidate responses, gain practical ideas for the classroom and put questions to our dedicated team by registering for one of our events here.	https://www.eduqas.co.uk/home/professional-learning/
Past papers	Access the bank of past papers for this qualification, including the most recent assessments. Please note that we do not make past papers available on the public website until 12 months after the examination.	Portal by WJEC or on the Eduqas subject page
Grade boundary information	Grade boundaries are the minimum number of marks needed to achieve each grade. For linear specifications, a single grade is awarded for the subject, rather than for each component that contributes towards the overall grade. Grade boundaries are published on results day.	For unitised specifications click here: Results and Grade Boundaries and PRS (eduqas.co.uk)

¹ Please note that where overall performance on a question/question part was considered good, with no particular areas to highlight, these questions have not been included in the report.

Exam Results Analysis	Eduqas provides information to examination centres via the WJEC Portal. This is restricted to centre staff only. Access is granted to centre staff by the Examinations Officer at the centre.	Portal by WJEC
Classroom Resources	Access our extensive range of FREE classroom resources, including blended learning materials, exam walk-throughs and knowledge organisers to support teaching and learning.	https://resources.eduqas.co.uk/
Bank of Professional Learning materials	Access our bank of Professional Learning materials from previous events from our secure website and additional pre-recorded materials available in the public domain.	Portal by WJEC or on the Eduqas subject page.
Become an examiner with WJEC.	We are always looking to recruit new examiners or moderators. These opportunities can provide you with valuable insight into the assessment process, enhance your skill set, increase your understanding of your subject and inform your teaching.	Become an Examiner Eduqas

Contents

	Page
Executive summary	5
Component 1 – Programming and System Development	6
Component 2 – Computer Architecture, Data, Communications and Application	10
Component 3 NEA – Programmed Solution to a Problem	13
Supporting you – useful contacts and links	16

Executive Summary

The Summer 2025 assessment series for GCE A Level Computer Science demonstrated a broadly positive performance across all components, with most candidates engaging well with the assessments and showing secure understanding of core technical content. The structure and balance of the assessment objectives (AOs) were maintained across Components 1 and 2, and the Non-Exam Assessment (NEA) in Component 3 continued to offer candidates the opportunity to showcase their programming and problem-solving skills in depth.

In Component 1, candidates performed strongly in procedural and logic-based tasks, particularly in areas such as Boolean algebra, truth tables and algorithmic reasoning. High attempt rates and facility factors in these questions reflected thorough preparation. However, extended descriptive responses, such as those evaluating system investigation methods or compilation stages often lacked depth and contextual application. Theoretical topics like Big O notation also proved challenging, indicating a need for greater emphasis on abstract reasoning and structured written responses.

Component 2 followed a similar pattern, with candidates excelling in well-practised topics such as binary arithmetic, floating-point representation and SQL. Descriptive questions, including those on network routing and data security were less well answered, often due to limited technical detail or lack of structured reasoning. Candidates would benefit from integrating examples and focusing on the technical aspects of such topics to improve clarity and depth.

The NEA (Component 3) revealed a wide range of candidate performance. Many centres supported the process effectively, with candidates producing well-structured investigations, detailed designs and functional prototypes. High-performing candidates demonstrated strong programming skills, effective use of validation and error handling and thoughtful evaluations. However, some candidates struggled to link their investigation to their final specification or included generic content not tailored to their project. Developmental testing was generally well evidenced, though some centres mixed it with final testing. Clear separation and traceability to objectives remain essential.

Across all components, common areas for improvement include the development of structured, example-driven descriptive writing, deeper understanding of theoretical concepts and clearer application of technical terminology. Centres are encouraged to make use of WJEC/EDUQAS' extensive range of resources, including blended learning materials, exam walk-throughs and CPD events to support teaching and learning in these areas.

COMPUTER SCIENCE

GCE A level

Summer 2025

COMPONENT 1 – PROGRAMMING AND SYSTEM DEVELOPMENT

Overview of the Component

The Component 1 examination is well established and produces consistent outcomes which result in stable grade boundaries.

The examination addresses each of the three Assessment Objectives (AOs) described in the specification, as follows: -

	Description	Weighting
AO1	Demonstrate knowledge and understanding of the principles and concepts of computer science, including abstraction, logic, algorithms and data representation.	44 /100
AO2	Apply knowledge and understanding of the principles and concepts of computer science, including to analyse problems in computational terms.	40 /100
AO3	Design, program and evaluate computer systems that solve problems, making reasoned judgements about these and presenting conclusions.	16 /100

The structure and balance of Assessment Objectives (AOs) in this Component 1 assessment were upheld, ensuring consistent coverage of the specification content. This series was generally well received, with the majority of candidates finding the paper accessible. Attempt rates ranged from 93.9% to 99.6%, exceeding 96% in most questions, indicating broad engagement across all areas.

Candidates demonstrated secure knowledge in procedural and logic-based tasks, with high marks awarded where responses required algorithmic reasoning, Boolean manipulation and structured programming constructs. For example, questions involving truth tables, search algorithms and Boolean algebra consistently produced strong outcomes, confirming thorough preparation in these technical areas.

As observed in previous sessions, responses to extended descriptive questions, particularly those requiring evaluation of systems investigation methods, changeover strategies, or compilation stages, often lacked depth, despite good engagement. While many candidates showed familiarity with key terms, the reasoning offered was frequently brief or unsupported by clear examples.

To maximise performance in these areas, candidates should focus on developing more structured and technically grounded responses. Incorporating relevant scenarios and applying terminology within a defined context will help ensure coverage across all AOs and enable access to the higher mark bands.

Comments on individual questions/sections

- Q.1** (a) This question required candidates to construct and traverse a binary tree. With a facility factor of 69.8% and a very high attempt rate of 99.6%, the task was accessible and generally well answered. The mean score of 4.2 out of 6 indicates many candidates understood tree rules and traversal methods, though some candidates misapplied alphabetical pointers or made minor ordering errors.
- (b) Asking for a description of situations where a binary tree is preferred saw lower success. The facility factor was 35.7% and 96.1% attempts. With a mean of 1.4 out of 4, responses were often vague or lacked technical justification. Despite strong engagement, answers lacked depth or real-world context.
- Q.2** This Backus-Naur Form construction question performed strongly with a facility factor of 73.6% and a 99.1% attempt rate. The mean score of 4.4 out of 6 reflects good understanding of syntax rules. Most candidates articulated production rules clearly though accuracy around uppercase constraints. Candidates were often able to access most of the marks, with only the date causing issues.
- Q.3** (a) Identifying the bubble sorting method scored very highly of a facility factor of 90% and 99.3% attempted. This confirms strong recognition of procedural structures and algorithmic patterns.
- (b) Dry running the sort algorithm showed a facility factor of 37.8%, despite a 97.2% attempt rate. The mean score of 2.3 out of 6 suggests procedural understanding may be superficial; errors emerged in table completion or swap logic after the first iteration, where the swap and counter were reset to 0.
- Q.4** (a) Constructing a truth table for a complex Boolean expression resulted in a facility factor of 74.6%, with 99.6% attempted. A mean of 3.7 out of 5 indicates candidates were confident with syntax and operator precedence.
- (b) Constructing a truth table for proving simple Boolean expression are alike produced the highest facility factor across the dataset at 95.3%, with a matching 99.6% attempt rate. Candidates clearly understood the logical equivalency and truth table construction.
- Q.5** (a) Boolean simplification was moderately well performed with a facility factor of 63.7%, 98.5% attempts, and a mean of 3.2 out of 5. Most errors related to misapplication of distribution or redundant steps.
- (b) The second Boolean simplification proved slightly more successful, with a facility factor of 66.3% and 98% attempts. Candidates generally applied the De Morgan's laws correctly, yielding a mean score of 2.7 out of 4.
- Q.6** (a) Describing shortest-path applications saw a facility factor of 58.3% and 97% attempts. The mean score of 1.2 out of 2 implies reasonable understanding of real-world uses such as network routing.
- (b) This multipart graphical and pathfinding question performed strongly, with a facility factor of 83.9% and 98.3% attempted. The mean of 4.2 out of 5 reflects confident identification and cost calculation once the network was visualised.

- Q.7** (a) Representing a queue in an array had a facility factor of 57.8% with 98% attempted. The mean of 1.7 out of 3 indicates that while candidates attempted the structure, issues arose with pointer placement or labelling.
- (b) An application-based question saw a high facility factor of 84%, with 97.8% attempts. Many candidates successfully referenced suitable examples such as printer queues or process scheduling.
- Q.8** (a) Evaluating efficiency and Big O notation scored lowest overall with a facility factor of 24%, 96.1% attempted, with a mean of 1.2 out of 5. Despite good engagement, many candidates were unclear on theoretical efficiency or misidentified the algorithm type.
- (b) Visual graph representation of time complexity was stronger, with a facility of 59.6% and 96.8% attempted. The mean score of 1.2 out of 2 implies basic recognition of growth trends.
- Q.9** (a) Methods of system investigation saw a facility factor of 76.1% and 97% attempt rate. The mean score of 4.6 out of 6 showed that candidates could explain interviews, observation and documentation review clearly.
- (b) Changeover methods performed slightly lower with a facility factor of 68.4% with 97.4% attempts. Many candidates could clearly describe parallel implementation with some level of detail although some lacked situational comparisons. On occasion, candidates incorrectly described phased implementation and instead gave a description of the pilot implementation method.
- Q.10** (a) Writing an unsorted search algorithm achieved a 73.6% facility factor and 95.2% attempt rate. The mean score of 3.7 out of 5 showed that candidates could formulate linear search structures accurately.
- (b) Stating advantages of linear search reached a facility factor of 86% with a 97.8% attempt rate, reflecting strong knowledge of its simplicity and reliability across unsorted arrays.
- Q.11** (a) Role of codes of conduct scored 60.7%, with 98.5% attempt. Mean score of 1.2 out of 2 shows most responses identified professionalism or ethical compliance, but depth was limited.
- (b) Extended discussion on computers' impact on employees saw lower performance with a facility factor of 46.3%, attempt rate 99.6%, and mean 3.7 out of 8. Answers often lacked detail, employee specific or technical reference.
- Q.12** The compilation process scored a facility factor of 52.3% with the lowest attempt rate at 93.9%. The mean score of 6.3 out of 12 reflects partial understanding, with many recalling stage names but struggling to apply them contextually.

Summary Observations

Procedural and structured questions, including truth tables, Boolean logic, sorting recognition and syntax rules consistently showed high facility and engagement, reflecting confident candidate preparation in core technical areas. Descriptive and extended responses, particularly those involving performance analysis, systems investigation or compilation stages were less successful. These responses showed good engagement but lacked depth or structured reasoning. Low facility in theoretical tasks like Big O evaluation confirms continued need to strengthen abstract reasoning and applied theory.

To improve outcomes, centres should focus on developing clear, example-driven descriptive writing and deeper understanding of computational theory, alongside the already strong practical fluency evident throughout the data.

COMPUTER SCIENCE

GCE A level

Summer 2025

COMPONENT 2 – COMPUTER ARCHITECTURE, DATA, COMMUNICATION AND APPLICATIONS

Overview of the Component

The Component 2 Examination is well established and produces consistent outcomes which result in stable grade boundaries. The examination addresses each of the three Assessment Objectives (AOs) described in the specification, as follows: -

	Description	Weighting
AO1	Demonstrate knowledge and understanding of the principles and concepts of computer science, including abstraction, logic, algorithms, and data representation.	44 /100
AO2	Apply knowledge and understanding of the principles and concepts of computer science, including to analyse problems in computational terms.	40 /100
AO3	Design, program, and evaluate computer systems that solve problems, making reasoned judgements about these and presenting conclusions.	16 /100

The balance between the AOs is set and adhered to, which, together with the scope of the specification content, produces a structure for a consistent assessment.

For this series, most candidates found the assessment to be accessible with 'attempted rates' ranging from 92.7% - 100% and exceeding 97% for most questions. The candidates were well prepared and were able to demonstrate a wide knowledge of the topics in the specification, with questions requiring precise responses to programming or mathematical problems producing high marks.

As in previous series the questions requiring more descriptive answers often produced responses that were correct but lacked adequate reasoning or detail. For these types of questions, it is suggested that candidates should concentrate on the technical aspects of the topic and include examples to illustrate and help structure their responses.

Comments on individual questions/sections

Example AO1 Questions

Q.1 (a) Explain, using suitable examples, the relationship between word size and clock speed. Mean mark of 1.9 out of 6 with a facility factor of 31.0.

Many candidates were able to describe clock speed, but fewer demonstrated a clear understanding of word length and explanations of the relationship between the two in terms of system performance, were often confused.

- Q.1 (b)** Describe the different functions carried out by RAM and Cache memory. Mean mark of 2.3 out of 4 with a facility factor of 58.5.

A well addressed part of the opening question with most candidates able to describe cache memory in terms of speed, location, and storage of frequently accessed data. The functions of RAM tended to be limited to the storage of data and program instructions, with few mentions of currently running processes or storage of the operating system.

- Q.2 (c)** Hexadecimal numbering system. Mean mark of 2.9 out of 6 with a facility factor of 48.3

This question comprised two parts - (i) Describe the relationship between hexadecimal and binary numbering systems and (ii) Describe two advantages of using hexadecimal numbering. Both parts of the question were addressed well by candidates able to demonstrate a clear understanding of the relationship between 4-bit binary numbers and hexadecimal digits.

- Q.6** Describe network routing and the role of routers in network communications. Mean mark of 1.8 out of 6 with a facility factor of 30.1.

This topic proved to be more challenging than expected, with many candidates providing a description of packet switching and the routers' role of forwarding data packets towards their destination address, but without mentioning path optimisation, or connections between LANs and WANs.

- Q.7 (b)** Describe the storage requirements of Boolean and string data types. Mean mark of 2.8 out of 4 with a facility factor of 69.5.

An AO1 item added to the AO2 database design question. Most candidates were clear about the nature and storage requirements of Boolean values and many candidates provided clear descriptions of a string data type, although details of the associated storage requirements were sometimes omitted or calculated wrongly.

- Q.12** Protection of the security and integrity of an organisations data. Mean mark of 4.5 out of 10 with a facility factor of 44.8.

An open topic within the context of an organisation that receives orders via the internet, stores data about customers and staff and has its own remote networked backup facility. The question was marked with a range of points identified in the indicative content of the mark scheme. Most candidates were able to describe a range of measures associated with cyber and physical security. Measures relating to data integrity, such as User Access Management and staff training were less well covered and very few candidates noted the requirement of the question to present a response in the form of a report.

Example AO2 Questions

- Q.2** Binary arithmetic and two's complement. Mean mark of 5.6 out of 6 with a facility factor of 93.3

An accessible and well-practiced topic, producing the highest facility factor of the assessment, with many examples of candidates achieving full marks.

- Q.3** Floating point numbers. Mean mark of 5.3 out of 6 with a facility factor of 88.3.

Also, an accessible and familiar topic with most candidates able to produce accurate conversions between decimal values and the specified floating-point forms.

- Q.4** (a) Assembly language – dry run code. Mean mark of 3.2 out of 4 with a facility factor of 78.8.

This is a well taught and practiced topic and most candidates were able to follow the code provided and produce the required outputs. Candidates who presented a tabulated response with columns for the registers and accumulator, and a row for each instruction often gained full marks.

- Q.7** (a) Database design. Mean mark of 4.4 out of 8 with a facility factor of 55.0.

Most candidates were familiar with the design work required and were able to interpret the given scenario to produce reasonable entity relationships. Data tables also tended to be well done, particularly where ERDs had been developed to consider key fields.

- Q.9** Explanation of operations using Index sequential files. Mean mark of 3.3 out of 8, with a facility factor of 41.3.

Many candidates had difficulty explaining the hierarchical relationships between and within different indexes. This appeared to be more a problem of clarity in the written work than of a lack of understanding of the sequences involved. Using a diagram to illustrate the stages involved in part (b) may have helped candidates to explain the process more clearly.

AO3 Questions

- Q.4** (b) Assembly. Language - write a program. Mean mark of 4.3 out of 6 with a facility factor of 71.5.

Most candidates were able to translate the algorithm provided and produce a functional program. Many examples gained full marks, where errors did occur, these tended to arise due to missing input or an incorrect loop structure.

- Q.8** SQL commands. Mean mark of 5.6 out of 10 with a facility factor of 56.0.

A well taught and practiced topic. Many candidates were able to produce the required commands correctly and gained full marks. Where errors did occur these most frequently involved omitting one of the search conditions for part (b) or not including 'VALUES' for part (c).

COMPONENT 3 – PROGRAMMED SOLUTION TO A PROBLEM

Overview of the Component

This component requires the learners to investigate, design, prototype, refine the design, implement, test and evaluate a computer solution to a substantial problem of their own choice. The learner's chosen problem must provide sufficient scope for them to access the marks available for each section of the work.

Learners need to investigate their chosen problems in sufficient detail to identify how data is collected, processed and output currently. The current system may be either paper-based or electronic.

Following the identification of their problems, learners should prepare sufficient documentation to allow them to take part effectively in the discussion with their teachers and/or peers.

Notionally this task will require 72 guided learning hours, which includes teaching time.

Tasks

Comments on tasks/questions relating to candidate performance/meeting assessment criteria

Discussion:

This section offers candidates the opportunity to present their proposed problem to their teacher, peers or other competent third parties. Candidates should receive detailed and informed feedback regarding the scope and suitability of their chosen problem and reflect deeply on this input. Reflection should result in refinements to their ideas, ensuring unsuitable topics are revised or replaced. Many centres have supported this process effectively, with candidates often enhancing the depth of their proposals or where appropriate, discarding their original ideas in favour of more viable ones—an approach that is highly effective and commendable.

Investigation:

Many candidates selected original problems that did not involve existing end users. To access the higher mark bands, candidates in this situation should research similar commercial systems, identifying common characteristics and features. A comprehensive working specification—complete with measurable objectives—should follow the research. However, in some cases, there is a disconnect between the investigation and the final specification, with candidates introducing new ideas not supported by their research.

Design:

The design phase allows candidates to plan and develop the technical aspects of their functional solution. This stage should be forward-thinking and include:

- Detailed screen layout designs
- Consideration of all required inputs and outputs
- Evidence of data structures and methods of access
- Consideration of appropriate data validation
- Program routine designs using recognised conventions (e.g., pseudocode or flowcharts)

Candidates should ensure that all elements of their design are relevant and clearly linked to their specific project. In some cases, candidates included generic descriptions (e.g., validation types or sorting algorithms) without applying them to their solution, which reduces their value. Most centres, however, provided excellent, detailed designs that could feasibly be implemented by a third party.

Prototype:

At this stage, candidates should demonstrate a partially complete system, self-reflect on progress and identify any changes in strategy required. The prototype should highlight key components of the system in development, with justification for the sections chosen. Many candidates effectively used screenshots or video evidence to show their prototype in action, video being particularly valuable for demonstrating system functionality. Submitting screenshots of code alone is discouraged, as it is difficult to ascertain the functionality of the submitted code.

Centres are encouraged to submit programming code that the moderator can run as evidence for this section.

Post prototype Refinement of Design:

Following third-party feedback, candidates should refine their design and provide justifications for accepting or rejecting suggestions. This section must focus on changes to the **design** and not include completed code. Centres that show "before and after" comparisons of designs provide strong evidence here. However, the inclusion of final programmed changes is inappropriate in this section and will not gain any credit, this should instead be assessed under Software Development.

Software Development:

Some centres have submitted outstanding work, showcasing a wide range of creative and effective solutions. To access the highest mark bands, candidates should:

- Fully exploit the programming language, using advanced features and normalised data structures (to 3NF)
- Demonstrate effective searching and sorting techniques
- Implement a system that meets all defined objectives and user requirements
- Include extensive input validation and error handling
- Use self-documenting identifiers and detailed annotations, making the code easy to follow for third parties

In some cases, code was difficult to execute due to missing packages or libraries. In such instances, candidates must include all necessary files or provide clear instructions in a README file.

Centres that included video demonstrations of the completed systems helped streamline the moderation process.

Developmental Testing:

Most centres presented detailed developmental testing throughout the software development process. Candidates showcased their problem-solving and decision-making skills effectively. However, in some cases, this testing was either absent or incorrectly merged with final testing. Developmental testing should be ongoing and clearly separate from final testing evidence.

Testing:

Centres that included well-structured test tables (with clearly identified **typical**, **extreme**, and **invalid** test cases) provided strong evidence. However, some candidates misunderstood **extreme testing**, which should involve inputs at the boundaries of allowable values—not beyond them.

Many centres supported this section with video evidence linked to each test. Where multiple tests were shown in a single video, timestamps were useful in guiding moderators to relevant sections.

Testing should focus on the functionality of the system in terms of:

- Input mechanisms, including validation for reasonable data entry
- Processing routines, ensuring correct and accurate output
- Output presentation, both on-screen and paper-based

Each test should be traceable to a specific objective. Candidates should provide thoughtful commentary and suggest refinements where test results do not meet expectations.

Evaluation:

It is encouraging to see many centres accurately assess evaluations. High-performing candidates provide detailed, reflective analyses of their work across all sections of the project. These evaluations include observations on performance, lessons learned, and improvements for future projects.

To achieve higher marks, evaluations must go beyond simply confirming that objectives have been met. While this can be included in the testing section, the evaluation should focus on critical self-reflection and analysis of the development process, including consideration of alternative approaches and personal development as a programmer.

Task marking

Comments on approaches to internal marking

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 Component 3 Task sheet. The centre should make notes on the Centre Mark Sheet the nature of any assistance given and the extent to which the solution actually works as stated in the report there should also be specific reference to the assessment objectives in the comments written on the work and coversheets.

Supporting you

Useful contacts and links

Our friendly subject team is on hand to support you between 8.30am and 5.00pm, Monday to Friday.

Tel: 02920 265 401

Email: CS@edugas.co.uk

Qualification webpage: [AS and A Level Computer Science | Eduqas](#)

See other useful contacts here: [Useful Contacts | Eduqas](#)

CPD Training / Professional Learning

Access our popular, free online CPD/PL courses to receive exam feedback and put questions to our subject team, and attend one of our face-to-face events, focused on enhancing teaching and learning, providing practical classroom ideas and developing understanding of marking and assessment.

Please find details for all our courses here: <https://www.edugas.co.uk/home/professional-learning/>

Regional Rep Team

Our regional team covers all areas of England and can provide face-to-face and online advice at a time which is convenient to you.

Get in contact today and discover how our team can support you and your students.

[Regional Support Team | Eduqas](#)

Eduqas Qualifications

We are one of the largest providers of qualifications for schools, academies, sixth form and further education colleges across England, offering valued qualifications to suit a range of abilities. Each and every one of our qualifications is carefully designed to engage students and to equip them for the next stage of their lives.

We support our education communities by providing trusted qualifications and specialist support, to allow our students the opportunity to reach their full potential.



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