



FIRST ASSESSMENT
SUMMER 2022

GCSE (9–1)

COMPUTER SCIENCE

J277 For first assessment in 2022

Assessment Story: exploring our question papers

Version 1



www.ocr.org.uk/computerscience



INTRODUCTION

Our GCSE (9–1) Computer Science qualification has been updated for first teach 2020, with first assessment in Summer 2022.

We have produced this guide to help you prepare your candidates successfully. In this guide, we share the story of our assessment approach and explore our question papers with you.

To update our qualification, we talked extensively with teachers. Teachers told us they **wanted stability** so we have made minimal changes to our qualification. We also worked closely with senior examiners (the majority of whom are practising teachers) and our in-house expertise. We **tested** our sample assessment material with students and incorporated their **feedback** into our question paper **design**.

Practical programming is an important part of GCSE Computer Science. This guide **demonstrates** how classroom practice relates to the questions your students will answer in the exam.

'The *minimal change* will make it *easier* to continue teaching confidently.'

'I was fearing the requirement for a separate extra exam, or a computer-based test. Fear **averted***.'*

Teacher quotes from Summer 2019 Network meeting



Question paper structure

GCSE (9–1) Computer Science consists of two examined components. Candidates must sit both exam papers.

The qualification is marked out of a **total of 160 marks**. The marks, duration and weightings are the same for both components, as shown below:

Component	Marks	Duration	Weighting
Component 1: Computer systems This component focuses on the theoretical understanding of Computer Science.	80	1 hour 30 minutes	50%
Component 2: Computational thinking, algorithms and programming This component focuses on the practical application of Computer Science: program design, writing, testing and refinement.	80	1 hour 30 minutes	50%

'**Consistency** in the current examined assessment will **help teachers** the most to plan, and build confidence in what they are delivering.'

Teacher quote from 2019 survey



mi- modifie
ALTROP Object
mod.mirrow
peration
Irror mod "MIRROR
Fror mod use x = T
Lrror_mod.use_y = Fa
Operation == "MTRD
prop_mod.use x = c
Trop mod.use y = T
Operation Fa
mirror mod use
rror_mod.use v = F
mod.use_z = T
ob solast 1
C ob.select=1
ntext.scene.object
"Solocted" + str/m

ects[one.na

OPERATOR CL

OCR Exam Reference Language and high-level programming languages

You told us that having flexibility to choose which programming language you teach was important to you. We therefore did not restrict the range of programming languages you can use.

Our approach gives you control over how you deliver programming within your classroom. It allows you to teach in multiple languages to help demonstrate programming techniques which may not exist in one specific language.

In Component 2 Section B, some questions require candidates to answer in either OCR Exam Reference Language or a high-level programming language.

All high-level programming languages have similar core constructs. OCR Exam Reference Language also adopts similar core constructs.

You can find the OCR Exam Reference Language in the specification. An example of how we apply core constructs in marking can be found in the J277/02 Sample Assessment Material mark scheme.

'*Minimal change*. Changes are **sensible**. Logical and sound decision making from OCR.'

Teacher quote from Summer 2019 Network meeting

Assessment Objectives and weightings

Every question tests one or more Assessment Objective. Assessment Objectives are defined by Ofqual and have required, associated weightings.

To aid your analysis of published question papers, our mark schemes indicate the Assessment Objective(s) targeted in each question.

	Objective	Weighting
AO1	Demonstrate knowledge and understanding of the key concepts and principles of Computer Science.	30%
AO2	Apply knowledge and understanding of key concepts and principles of Computer Science.	40%
AO3	 Analyse problems in computational terms: to make reasoned judgements to design, program, evaluate and refine solutions. 	30%

Component 1: Computer systems

Component 1 consists of AO1 and AO2 questions only.

Component 2: Computational thinking, algorithms and programming

Section A tests **AO1**, **AO2** and **AO3**. Section B tests **AO3** only.



Component 1

Component 1 continues to follow the J276 model. There will be a mixture of short and medium answer questions as well as one 8-mark extended response question.

Component 2

Teacher feedback highlighted the benefit of clear sign posting for candidates about what is being assessed. Therefore we have divided Component 2 into two sections.

The breakdown of marks between both sections can be seen below:



Section A: 50 marks

In this section, candidates draw on their knowledge and understanding of concepts of computer science. Candidates then apply these to problems in computational terms, where they may use an algorithmic approach.

Candidates will be provided with broad contexts. They will analyse these in detail by deconstructing a problem into component parts, and then make links and connections between different strands of knowledge and understanding.

Section A will consist of short and medium answer questions with a points-based mark scheme. It will also always contain one 6-mark algorithm question. It was felt that a points-based mark scheme suited the algorithmic nature of these questions.

Candidates have flexibility and choice in how they present their answers in this section:

Questions asked in:	Candidates respond in:
Natural English	✓ Pseudocode
OCR Exam Reference Language	✓ Flowcharts
Flowcharts	✓ Bullet points
	✓ OCR Exam Reference Language
	✓ A high-level programming language

This flexibility allows us to focus on assessing a candidate's ability to structure answers logically without a focus on syntactic precision.

Section B: 30 marks

Section B will test a candidate's Practical Programming skills and their ability to 'design', 'write', 'test', and 'refine' programs. It will also test a candidate's ability to respond in a precise manner when creating algorithms.

Each question builds on the one before, although the questions are not dependent on each other.

Section B has been designed to mimic a real-life programming experience as far as possible under examination conditions. Section B assesses AO3 only. AO3 lends itself to scenario-based problem-solving questions.

Candidates will draw on their knowledge and experiences from across the full course of study and in particular their experience of programming within the classroom environment.

We have scaffolded Section B to build the candidate's confidence as they begin this new section. Lower demand questions are at the start.

The table below shows how we will assess a candidate's ability to 'design', 'write', 'test', and 'refine' programs.

Question Focus	Questions asked in:	Candidates respond using:
Design	Natural English	 ✓ Pseudocode ✓ Flowchart ✓ Tick box responses ✓ Natural English
Write	Pseudocode Natural English Flow chart	 ✓ OCR Exam Reference Language ✓ High-level programming language
Test	OCR Exam Reference Language	 ✓ Trace tables ✓ Test plans ✓ Tick box responses ✓ Natural English
Refine	OCR Exam Reference Language	 ✓ OCR Exam Reference Language ✓ High-level programming language ✓ Natural English

Find an example of a 'write ' question on page 17

'Write' questions will require candidates to answer either in OCR Exam Reference Language or the high-level programming language they are familiar with. Answers that are written in pseudocode, structured English or bullet points will not be given marks.

'Refine' questions may ask candidates to describe changes they would make to a program and therefore responses may be written in natural English. Other **'refine'** questions will require the use of OCR Exam Reference Language or a high-level programming language.



'Design' questions require candidates to think independently to come up with a proposal or solution to a brief.

'Test' questions still require candidates to use programming skills developed in the classroom. These questions will assess a candidate's ability to read and understand precise coding language.

Questions which require the use of OCR Exam Reference Language, or a high-level programming language will be clearly marked.

In these questions the requirement for precision will help test a candidate's experience of using a programming language as part of the course. Marks will be given for correct use of core constructs which are common across all programming languages.

Candidates will not be penalised for minor errors in their programming language syntax, e.g. a missing colon. This is to reflect real life practical programming because an IDE would automatically alert them to some errors and often suggest solutions to these.



Synopticity

Synoptic assessment tests candidates' understanding of connections between different elements of a subject. It involves the explicit drawing together of knowledge, skills and understanding within different parts of the course. For example, while the focus on assessing programming skills is in Component 2, candidates may also draw upon the knowledge, skills and understanding from Component 1.

Candidates will also benefit from a wider understanding of the impacts of Computer Science within society and the emerging development of technology. This experience can help candidates relate examination questions to real-life issues.

Showing working

If a candidate's final answer is correct, full marks would usually be given.

However, some questions specifically state to show any working out. This is often to assess the process/thinking of the candidate. A common example of this could be converting from hexadecimal to binary.

Candidates can still gain marks in questions that require working to be shown, even if the final answer is incorrect. On that basis it is good practice to always show working. That way it may still be possible to score some marks.



Accessibility Principles

The principles and rationale underpinning our paper construction approach for OCR GCSE Computer Science are outlined below. These make sure we're always assessing understanding of computer science without letting the language or formatting of our questions be an obstacle for understanding what is needed.

Group	No.	Accessibility principle	Why?
	1	 Arial font will be used except when we write programming code which will use Courier New. 	Courier New is used for programming language to ensure the spacing is identical between characters.
		 Adequate space for responses and room for working in calculations. 	To make it easy for candidates to add their responses/do their working.
Look and feel of the paper	2	 Tone (assessing good understanding of GCSE Computer Science without letting the language of our questions be an obstacle to understanding what is needed) The use of overly complicated language and grammatical constructions will be avoided. Contexts and vocabulary will be considered for currency and appropriateness to students, e.g. glasses not spectacles, dice not die, formulas not formulae. Language used throughout the question will be consistent. For example, usage in the stem of a question matches that throughout the rest of the question and any titles given to any diagrams. Technical words will be used appropriately to underpin the GCSE Computer Science being assessed. 	To make it as clear as possible what response is expected.
pproach	3	Command words will only ever be used from the defined list of command words for GCSE Computer Science.	To ensure clarity as to what can be assessed and how all command words will be used.
Assessment a	4	Negative questions will be kept to a minimum.	Used well, negative questions can be a good way of testing understanding but can also easily lead to confusion. We will only ever use negatives where it is the most appropriate approach.

Group	No.	Accessibility principle	Why?
pproach (continued)	5	Where there is a large context provided (e.g. a scenario) in a question, sentences will be grouped by content rather than be lots of separate sentences. Bulleted lists or numbering will be used where it helps indicate stages in a process/practical method.	To ensure information is presented in the clearest possible way.
Assessment ap	6	Names will not be used unless avoidance of names leads to a complicated question layout.	To avoid imparting cultural/ gender bias into questions through choice of name or confusing candidates through choices of names they are unfamiliar with.
ting style	7	All text will be left aligned (text in table headings will be centred except for row headings, which will be left aligned).	To align with the principles applied to our modified question papers (left alignment is easier to understand for a range of visual impairments).
Question format	8	Italics will not be used in questions. Generally italicised Latin abbreviations such as i.e., e.g. and etc. will not be used. English terms will be used instead.	Italics can be hard to read if overused.
		If a specific word requires emphasis, bold font will be used.	
	9	If a question requires an answer to a certain number of decimal places, for example, we will always ensure this is clearly stated.	To avoid confusing candidates who may be concerned about the required precision needed.
	10	Images, diagrams and data will only be used where they genuinely support what is required in the question. We will avoid candidates needing to turn pages by aiming to always have any images or diagrams required for a question on the same page or the facing pages.	To avoid unnecessary page turning and distracting images for the candidates that do not help them understand what is required in the question.
	11	All tables, graphs, images and diagrams will be left aligned.	To align with the principles applied to our modified question papers (left alignment is easier to understand for a range of visual impairments).
	12	Text will not be wrapped around images/diagrams/graphs.	To retain clarity.
	13	If candidates are required to do something with an image/ diagram/graph, it will be centred with sufficient space around it for them to do their working.	To avoid candidates struggling to fit in their response.



Command words

We have defined command words for use within J277 GCSE (9–1) Computer Science. Command words are used consistently throughout the examination papers. This allows:

- ✓ consistent understanding of the style of response required in a question
- ✓ clarity for teachers in understanding assessment purpose
- ✓ support for teachers when designing internal tests
- \checkmark familiarity and consistency within an examination paper
- ✓ clarity for candidates in the expectations of an examination question.

The exact response expected to a command word will be dependent on the context. We advise candidates to read the full question carefully to be sure of what they are being asked to do.

Command word	Definition
add	Join something to something else so as to increase the size, number, or amount.
analyse	Break down in order to bring out the essential elements or structure. To identify parts and relationships, and to interpret information to reach conclusions.
annotate	Add brief notes to a diagram or graph.
calculate	Obtain a numerical answer showing the relevant stages in the working.
compare	Give an account of the similarities and differences between two (or more) items or situations, referring to both (all) of them throughout.
complete	Provide all the necessary or appropriate parts.
convert	Change the form, character, or function of something.
define	Give the precise meaning of a word, phrase, concept or physical quantity.
describe	Give a detailed account or picture of a situation, event, pattern or process.
design	Produce a plan, simulation or model.
discuss	Offer a considered and balanced review that includes a range of arguments, factors or hypotheses. Opinions or conclusions should be presented clearly and supported by appropriate evidence.
draw	Produce (a picture or diagram) by making lines and marks on paper with a pencil, pen, etc.
evaluate	Assess the implications and limitations; to make judgements about the ideas, works, solutions or methods in relation to selected criteria.
explain	Give a detailed account including reasons or causes.

Command word	Definition
give	Present information which determines the importance of an event or issue, or to show causation.
how	In what way or manner; by what means.
identify	Provide an answer from a number of possibilities. Recognise and state briefly a distinguishing factor or feature.
justify	Give valid reasons or evidence to support an answer or conclusion.
label	Add title, labels or brief explanation(s) to a diagram or graph.
list	Give a sequence of brief answers with no explanation.
order	Put the responses into a logical sequence.
outline	Give a brief account or summary.
refine	Make more efficient, improve, modify or edit.
show	Give steps in a derivation or calculation.
solve	Obtain the answer(s) using algebraic and/or numerical and/or graphical methods.
state	Give a specific name, value or other brief answer without explanation or calculation.
tick	Mark (an item) with a tick or select (a box) on a form, questionnaire, etc. to indicate that something has been chosen.
what	Asking for information specifying something.
write/ rewrite	Mark (letters, words, or other symbols) on a surface, typically paper, with a pen, pencil, or similar implement/write (something) again so as to alter or improve it.



Question type examples

We use a range of different question types to allow broad specification coverage, helping us keep our examinations at a manageable length.

Types of question possible:

	Short answer	Medium answer	Extended response
Component 1	\checkmark	\checkmark	(one 8-mark question)
Component 2 Section A	\checkmark	\checkmark	×
Component 2 Section B	\checkmark	\checkmark	×

Short answer questions

Short answer questions may:

- be 1, 2, or 3-mark questions
- test any Assessment Objective (AO1, AO2, AO3).

Short answer questions allow us to:

- ✓ have broad specification coverage
- ✓ keep our examinations at a manageable length.

We have increased the number of tick box, multiple choice and 'fill in the blank' questions we use in our examinations. This allows candidates to show knowledge and understanding in a range of response styles.

These types of questions also allow candidates to show understanding of both simple and more complex ideas, without necessarily requiring a written response, or multi-line answer.

Examples of short answer questions

From our Sample Assessment Material Component 1 question 3(a)

3 A satellite navigation system (Sat Nav) uses RAM and ROM.

(a) Tick (✓) one box in each row to show whether each of the statements is true for the RAM or ROM in a Sat Nav.

	RAM	ROM		
Stores the boot up sequence of the Sat Nav.				
The contents are lost when the Sat Nav is turned off.		Tick box	Tick box style approach. Testing	
Holds copies of open maps and routes		context tick, AO	, one mark for each correct 2.	

From our Sample Assessment Material Component 2 Section A question 1(a)



identify three devices, AO1.

Medium response questions

Medium response questions may:

- ✓ be 3, 4, 5, or 6-mark questions
- ✓ test any Assessment Objective (AO1, AO2, AO3).

Component 2 Section A will always contain **one** 6-mark algorithm question. This will be marked using a point-based mark scheme. The algorithmic nature of these questions and responses suit this style of mark scheme, rather than a levels of response mark scheme.

Examples of medium response questions

From our Sample Assessment Material Component 2 Section A question 4(b)

(b)	The	e program design is updated to create usernames as follows:			
	•	If the person is a teacher, their username is the last 3 letters of their surname and then the first 2 letters of their first name. If the person is a student, their username is the first 3 letters of their first name and then the first 2 letters of their surname.			
	(i)	What would be the username for a teacher called Fred Biscuit using the updated process?			
	(ii)	Write an algorithm for the updated program design shown in question 4(b)(i) .			
		Testing AO3. As this is in Section A of Component 2, candidates can respond in pseudocode as they are			
		precision of language.			
		[6]			

For example, from our Sample Assessment Material Component 2 Section B, question 8(b)

(b) The program should only allow values from 0 to 300 inclusive as valid inputs. If the data entered breaks this validation rule an error message is displayed.	
(i) Complete the following program to output "Invalid input" if the data does not	
<pre>meet the validation rule. You must use either: OCR Exam Reference Language, or a high-level programming language that you have studied. mins = input ("Enter minutes played: ")</pre>	We make it clear that candidates are required to respond in OCR Exam Reference Language or a high-level programming language and will always use this same wording. This question tests the Practical Programming skill to 'write'. We will use OCR Exam Reference Language to ask these questions.
if mins < 0 mins	then
("Invalid input")	
endif	
	[3]





Extended response question

Component 1 will always have **one** extended response question worth 8-marks. Extended response questions test a candidate's ability to construct and develop a sustained line of reasoning.

Extended response questions are marked using a tiered mark scheme. Candidates are placed initially in 'bands' and then given a mark within that particular band.

For example, from our Sample Assessment Material Component 1 Question 11.

11 * People often want to buy the most up-to-date smartphones, even though the smartphone they own still works.

Discuss the impact of people wanting to upgrade to the latest smartphone.

In your answer you might consider the impact on:

.....

- smartphone users
- cultural issues
- ethical issues
- environmental issues

This question requires candidates to engage with the context and using their knowledge and understanding to weigh up and consider both sides of a discussion. Candidates need to present their answer in a logically structured format.

.....[8]

Looking to the future

When making our updates for first teach 2020, minimal change was important to teachers.

The introduction of Computer-Based Testing was discussed but many teachers said implementing this in the short term was not practical for a range of reasons. They also told us that they would be keen for OCR to explore Computer-Based Testing for the future.

We are therefore committed to doing this and will be continuing to research and explore this option and will be consulting with teachers further.



Meet the team

Our Assessment Standards Senior Manager looks after the live assessments, meanwhile our team of Subject Advisors provide day-to-day customer focused support.

Ceredig Cattanach-Chell

Subject Advisor

Joanne Plester Assessment Standards Senior Manager

Rajesh Sood Subject Advisor

Vinay Thawait Subject Advisor

OCR Resources: the small print

OCR's resources are provided to support the delivery of OCR qualifications, but in no way constitute an endorsed teaching method that is required by OCR. Whilst every effort is made to ensure the accuracy of the content, OCR cannot be held responsible for any errors or omissions within these resources. We update our resources on a regular basis, so please check the OCR website to ensure you have the most up to date version.

This resource may be freely copied and distributed, as long as the OCR logo and this small print remain intact and OCR is acknowledged as the originator of this work.

Our documents are updated over time. Whilst every effort is made to check all documents, there may be contradictions between published support and the specification, therefore please use the information on the latest specification at all times. Where changes are made to specifications these will be indicated within the document, there will be a new version number indicated, and a summary of the changes. If you do notice a discrepancy between the specification and a resource please contact us at: resources.feedback@ocr.org.uk.

OCR acknowledges the use of the following content: page 2: hands on keyboard/ESB Professional/Shutterstock.com; books and apple icon - PPVector/Shutterstock.com; page 4: programming code/ White Mocca/Shutterstock.com; page 8: students in the computing classroom/YanLev/Shutterstock.com; page 9: Monkey Business Images/Shutterstock.com; page 13: teacher and student/Monkey Business Images/Shutterstock.com; page 19: teachers/fizkes/ Shutterstock.com.

Whether you already offer OCR qualifications, are new to OCR, or are considering switching from your current provider/awarding organisation, you can request more information by completing the Expression of Interest form which can be found here: www.ocr.org.uk/expression-of-interest

Please get in touch if you want to discuss the accessibility of resources we offer to support delivery of our qualifications: resources.feedback@ocr.org.uk

Looking for a resource?

There is now a quick and easy search tool to help find **free** resources for your qualification:

www.ocr.org.uk/i-want-to/find-resources/

Need to get in touch?

If you ever have any questions about OCR qualifications or services (including administration, logistics and teaching) please feel free to get in touch with our **Customer Support Centre**.

General qualifications

Telephone 01223 553998 Facsimile 01223 552627 Email <u>general.qualifications@ocr.org.uk</u>

www.ocr.org.uk

OCR is part of Cambridge Assessment, a department of the University of Cambridge. For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored.

© **OCR 2019** Oxford Cambridge and RSA Examinations is a Company Limited by Guarantee. Registered in England. Registered office The Triangle Building, Shaftesbury Road, Cambridge, CB2 8EA. Registered company number 3484466. OCR is an exempt charity.



