Summary of the Examiner's Report 2018

OCR GCSE Computer Science J276

The following gives an overview of some of the key points and areas for students to improve on for the examination based on feedback from the 2018 examination series. This is not a replacement for the full examiner's reports which are highly recommended for teachers to read.

Legal issues
Unit 4 Ethical, legal, cultural and environmental concerns

Copyright law and computer systems

- Computer programs are subject to copyright law
- A user interface, and the 'look and feel' of an operating system, can also be subject to copyright law
- In 1988 Apple launched a $5.5 billion law suit against Microsoft and Hewlett-Packard for copying their GUI interface, first developed in the 1980s. In 1993, they finally lost the case

Ethical and cultural concerns
Unit 4 Ethical, legal, cultural and environmental concerns

Artificial intelligence

- The same techniques used by AlphaGo can be used to teach computers to:
    - recognise faces
    - translate between languages
    - show relevant advertisements to Internet users
    - drive cars

OCR
Oxford Cambridge and RSA
An OCR endorsed teaching and learning tool

GCSE OCR
Computer Science
J276

Ethical, legal, cultural and environmental concerns

Unit 4

4

PG ONLINE

Computers in the modern world
Unit 4 Ethical, legal, cultural and environmental concerns

Security concerns

- GCHQ, the British Government intelligence and security organisation, has intervened in the design of an £11bn nationwide system of smart meters
- It feared that the lack of security would enable hackers to cause a collapse in the National Grid

Ethical and cultural concerns
Unit 4 Ethical, legal, cultural and environmental concerns

Privacy issues

- Every time you log on to a web site, data about your visit may be collected and stored
- Every time you use your phone to make a call, data about the number you called, the time and date you called, and the duration of your call, may be collected and stored
    - Is this acceptable?

One of our free editable teaching units for OCR GCSE Computer Science J276

Boundary data:

J276/01 Computer Systems

80 marks. Grade boundaries – June 2018

Grade	9	8	7	6	5	4	3	2	1
Mark required	68	62	56	49	42	35	26	18	10
% required	85%	78%	70%	61%	53%	44%	33%	23%	13%

All percentages are rounded to the nearest percent

J276/02 Computational thinking, algorithms and programming

80 marks. Grade boundaries – June 2018

Grade	9	8	7	6	5	4	3	2	1
Mark required	68	61	55	48	41	34	26	17	9
% required	85%	76%	69%	60%	51%	43%	33%	21%	11%

All percentages are rounded to the nearest percent

J276 - Overall

Total marks available for the two examinations: 160
Grade boundaries – June 2018

Grade	9	8	7	6	5	4	3	2	1
Mark required	136	123	111	97	83	69	52	35	19
% required	85%	77%	69%	61%	52%	43%	33%	22%	12%

All percentages are rounded to the nearest percent

This is the first examination series available, so no previous data is available.

Previous grade boundaries – 2015-17

Average grade boundaries for the previous specification's examinations were as follows:

Grade	A*	A	B	C	D	E	F	G
Average '15-'17	77%	67%	56%	46%	37%	28%	20%	12%

> 12% of students improved their GCSE grade in schools using PG Online GCSE Teaching Units

Sample: 59,599 Computer Science students, 2018

Improve results

Save hours of planning

J276/01 Computer Systems

Key information

Marks available	80
Calculator use	Not allowed
Time allowed	1h 30m
Resources	Black ink / ball-point pen
Question choice	Answer all questions

1. Students need to be aware of command words (see section 5e in the specification). If 'describe' or 'explain' questions are given they need to expand their answers
2. In 'Quality of Written Communication' questions, students need to cover a number of points and discuss both the positives and negatives
3. Full answers should be given to questions – not just key words
4. Students should make their answers match the context of the question
5. Algorithm questions require an actual algorithm not a repetition of the question
6. Many students have misconceptions that secondary memory (e.g. hard drive) is required due to insufficient storage in primary memory or to backup data rather than so that data is stored when the device is turned off
7. Explain questions such as 'explain why this is the most appropriate...' do not require a just a list of benefits. Instead they should identify the benefits and then expand each one and apply to the scenario or context
8. An explain question that first introduced networks and then in the sub question asks 'explain what is meant by a protocol' would first require the meaning of protocol (a set of rules) and then expand the answer to the context of the question (networks)
9. If a question explicitly asks for an algorithm to be written in pseudocode, then it will not gain marks if it is written as a flowchart. Equally, a question that asks for an algorithm to be written as a flowchart will not gain marks if answered with pseudocode
10. Students who have difficulties with algorithm questions should remember that they will gain marks (where appropriate) for input and output statements
11. The statement `INPUT = variableName` will not gain marks in pseudocode as the variable name needs to be on the left of the assignment operator. E.g. `variableName = INPUT`. `INPUT variableName` is an acceptable alternative.
12. String concatenation is not enough for an output e.g. `print(hello + name)` – the string must be in quotes, e.g. `print("hello " + name)`
13. A use of x instead of * in pseudocode would still get marks
14. Generic answers are not sufficient. E.g. If a question asks for a description of the function of a router, an answer 'it connects devices together' is not sufficient. Instead answers should describe how routers are used to receive packets from computers, read the destination address of each and then forward each packet to its destination
15. Many students are not familiar with DNS/Domain Name Servers and Domain Name System for converting URLs to IP addresses
16. Some students have the misconception that virtual memory is slower as it is physically further from the processor. It is slower as it must first be transferred to RAM.

Note: This summary is the interpretation of PG Online and has not been through any accreditation process by the examination board.

Improve results

Save hours of planning

J276/02 - Computational thinking, algorithms and programming

Key information

Marks available	80
Calculator use	Not allowed
Time allowed	1h 30m
Resources	Black ink / ball-point pen
Question choice	Answer all questions

1. Be aware that content on SQL can be assessed in this section in addition to algorithms with pseudocode and/or flowcharts
2. SQL that needs to be learnt by students is given in section 5f of the specification - **SELECT**, **FROM**, **WHERE**, **LIKE**, **AND**, **OR** and the wildcards ***** and **%**
3. Pseudocode used in examinations is also given in section 5f. The pseudocode students write does not need to match this syntax as long as it can "be reasonably inferred by a competent programmer"
4. Students need to be familiar with the mathematical symbols for logical operators ∧ (AND), ∨ (OR) and ¬ (NOT). Whilst the examination may use these symbols, students are not expected to use them in their answers and AND/OR/NOT will be sufficient if required
5. Arrays will always start at zero. Many students reference the wrong elements of the array as they incorrectly start with the first element being 1
6. Most students understand how count-controlled loops/iteration works, however, most do not understand how nested loops work – i.e. that the nested loop completes fully for each iteration of the outer loop
7. The basic constructs that students should be aware of are "sequence", "selection" and "iteration". "Loops" will be accepted instead of "iteration"
8. Marks have been given for the ability of a variable to be changed. This is generous and students should be reminded that it is the **value stored** that can be changed
9. In pseudocode, `input("enter name")` will not gain marks as the result needs to be assigned to a variable to store it – e.g. `name = input("enter name")`
10. Equally, two values cannot be input at the same time as a variable will only store one value. Instead, use `a = INPUT("Enter a")` then `b = INPUT("Enter b").` `INPUT a, b` would be an acceptable alternative.
11. Most students understand that by using subroutines/functions/procedures code does not need to be copied and pasted many times. However, when asked to use these in pseudocode they then copy and paste code rather than calling a function. Many students need more practice with subroutines in programming/pseudocode
12. Students struggled with a simple question that looked at while loops, for loops and whether they would result in infinite loops. More practice of these is advised
13. A count-controlled loop is a FOR loop. When the question requested a count-controlled loop, many students gave a WHILE loop in their answer which was incorrect (even if it incremented a variable inside the loop). Those students who study Python may benefit from seeing how another language deals with loops such as Visual Basic (which has closer FOR loops to the pseudocode in the specification)
14. A common error in IF statements is if `name != "Sam" or "sam"`. This should be: `if name != "Sam" or name != "sam"`
15. Candidates should be careful with quotes around strings. E.g. `choice = A` (which assigns a variable) is very different to `choice = "A"` (which assigns a value).