# GCSE

# Computer Science

Paper 1

Report on the Examination

8520

June 2018

Version: 1.1

Further copies of this Report are available from aqa.org.uk

## Introduction

This year saw the first set of examinations for the reformed GCSE Computer Science specification, with some areas being examined for the first time. The students, on the whole, seemed well prepared for this examination paper and showed good application of the problem solving and computational thinking skills they have been taught.

Key points to take forward from this year's examination paper would be to ensure students read the question carefully as some student responses indicated that this was not always happening. Where calculations were expected to be shown, students could still gain marks for showing their working even when they ended up with incorrect final answers. Unfortunately some students were not showing their working and so were not able to gain any marks. They should always be encouraged to show their working for this reason.

The algorithm questions played a big part in this paper, requiring students to create an algorithm using pseudocode or a flowchart. Following the same requirements as the legacy specification students are free to express their algorithms using either notation and were rewarded for clear and unambiguous instructions; mistakes in the use of flowchart symbols are normally ignored, except where they make the algorithm unclear. An example would be unannotated arrows emerging from a decision symbol.

One positive aspect of many pseudocode solutions seen was the good use of indentation which frequently made the algorithms easier to read and follow the structure of.

## Question 1

This question was about key terms used in computational thinking.

Most students did well in this section and it was pleasing to see a good understanding of new key terms such as abstraction and decomposition which haven't been tested before.

## Question 2

This question was about identifying parts of an algorithm.

The first 4 parts were answered well and again students showed a good understanding of key terms. However, more than half were unable to trace through the algorithm for part 2.5 but many could identify where the error in the algorithm was and how to fix it.

## Question 3

This question was about binary arithmetic.

This was another new topic for the reformed specification and whilst the majority of students showed their knowledge of how to shift a binary number, far fewer were able to describe the arithmetic effect, with many choosing to show the binary answer instead, which is not what the question asked them to do.

## Question 4

This question was about sound conversions and file sizes.

Students found this question challenging with less than 25% getting the full 2 marks for part 4.1, with many only giving one part of the answer and some describing how sound is sampled which again was not what the question was looking for.

The calculation question was answered well with more than half the students gaining the full 4 marks available; those who didn't get the correct answer still managed to get some working marks if they had shown their calculation. Common errors included not dividing by 8 or having the correct calculation but arriving at the answer 500 instead of 5000.

Less than half of the students were able to identify that 5 bits will allow 32 bit patterns the most common mistake being to double the 4 bits to 8 bits.

**Question 5**

This question was the first of the questions on this paper requiring the students to create an algorithm using pseudocode or a flowchart.

It was pleasing to note that over a quarter of students received full marks on this question and well over half scored at least 4 out of 7 marks. The mark scheme shows the breakdown of the seven marks – students who find this type of question difficult should note that in this case one mark is awarded for using input and another for storing it as a variable so `character ← USERINPUT` was worth two marks. Students also need to avoid rewriting the question in process boxes as we have seen an increase in this type of answer this year and it is unlikely to gain the student any marks.

**Question 6**

This question was the second of the questions where the students need to work with a given algorithm and also introduced sorting algorithms which are new to the specification.

Pleasingly 70% of the students were able to identify the Boolean datatype and over 75% managed to get one of the two marks available for identifying the need for meaningful variable names with only a quarter gaining the second mark. Students should be reminded that for a 2 mark question examiners are looking for two clearly distinct points.

The trace table caused some students issues with only a third of the students gaining 4 marks or above. The most common mistake with this trace table was students not repeating the sequence in column `i` twice.

The merge sort question was answered well with 75% of students scoring the full 3 marks available. This was pleasing to see, as this was a new topic for this specification. Following on, nearly 50% were able to say why this might be seen as a better algorithm than a linear sort and over 70% were able to give at least one reason for implementing a subroutine. However, many students gave the answer "quicker" or "faster" for part 6.6 without giving a reason why it is quicker or faster. At GCSE level this is too vague to pick up a mark and students need to ensure that they are qualifying their answers.

**Question 7**

This question was the second of the questions on this paper requiring the students to create an algorithm using pseudocode or a flowchart.
Over a quarter of students received full marks on this question and over 75% scored at least 4 out of the 8 available marks. The mark scheme shows the breakdown of the eight marks but many students made errors in the Boolean condition (forgetting to check for the case when the value is equal to 0). Another common error was the use of selection instead of iteration. There were three marks available for doing the 2 calculations and outputting the answer and a lot of students picked up marks here.

Again we saw an increase in students rewriting the question in process boxes which as stated earlier is unlikely to gain the student many if any marks.

**Question 8**

This question was about logic gates, logic circuits and truth tables.

The vast majority of students were able complete the truth table for the AND gate and nearly half managed to correctly draw the logic circuit given. Common errors on this question involved incorrect symbols being used for the logic gates and putting the NOT gate at the end of the circuit. The vast majority of students managed to pick up at least one mark on this question

**Question 9**

Question 9 was the blocks question with 3 sets of algorithms to work through and decide on the final positions of a set of blocks.

The first part was well answered, with over 60% of students correctly identifying the final positions of the blocks. When the WHILE loop was introduced in part 2, the number gaining the full 3 marks dropped slightly to 45% and with the FOR loop only 17% were able to correctly position the blocks. (The most common mistake was having the B and C blocks in column 0). Students have access to the AQA Pseudocode guide and should familiarise themselves with how the exam questions will be written as it seems that some were unprepared for how it worked.

Part 4 of this question asked for an algorithm to be created to move the blocks in order into the centre column. The question asked for the algorithm to work for any number of blocks but a lot of students only moved the blocks they were given. Interestingly a number of students reused the algorithm from part 9.2 and moved the last block over rather than detecting when the column was empty. Students were not penalised for this as the algorithm would still work with any number of blocks. However, on the whole, this part of the question was poorly answered in relation to the other two algorithm questions with just over 50% gaining any marks at all.

**Question 10**

The final question on the paper again asked the student to work through a given algorithm.

Students seemed to find this smaller trace table easier to complete than the one in question 6 with 60% of students gaining two or three of the available marks. In common with the previous specification the most common mistake when completing trace tables is still students not carrying the value in a while loop through to iterate to the final value (in this case 3).

Over half of the students could state the return value of the subroutine as being false and over 85% were able to identify the correction needed.

The last question asked the students to identify why changing a line of the algorithm could make it more efficient. Only just over 40% were able to identify at least one of the two available marks with common answers being “it’s faster” or “uses less storage”. These were not valid creditworthy answers.

## Mark Ranges and Award of Grades

Grade boundaries and cumulative percentage grades are available on the Results Statistics page of the AQA Website.

## Converting Marks into UMS marks

Convert raw marks into Uniform Mark Scale (UMS) marks by using the link below.

UMS conversion calculator