AQA

AS LEVEL
COMPUTER SCIENCE

7516/1 Paper 1 (Options A - E)
Report on the Examination

7516/1
June 2023

Version: 1.0

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

General

Most students showed good preparation for the exam, particularly in the questions requiring understanding of the skeleton program. There were many instances of scripts with no attempt at questions towards the end of the paper, although many students often picked up the more accessible marks in the larger programming tasks.

Students will not receive marks for captures of screens that have not been produced by running their code. Students should also make sure, when pasting in screen captures, that they are readable. The height of a capital letter should be at least 2mm when the EAD (Electronic Answer Document) is printed. This is a skill that should be practiced before the exam, using an EAD and a past paper.

When the evidence requested is an amended subroutine, the entire edited subroutine should be added to the EAD. Full credit can often not be awarded if the examiner cannot ascertain the location of any new code. The code should be copied and pasted as text into the EAD, rather than as a screen capture – there is no need for any IDE formatting to be shown and screen captures can often miss the end of lines.

When a question requests an identifier of a variable or a subroutine, no other code should be included. If the answer to a question is the subroutine name DisplayMenu, an answer of DisplayMenu() would not be creditworthy. The parentheses are not considered part of the identifier, nor would the data type be considered part of a variable's identifier.

Question 1

Most students secured full marks in this question, which required an understanding of a state transition diagram. Of those who did not secure full marks, a minority provided an incomplete answer, but one that was correct as far as it went. Although the question noted that some states required multiple labels this was missed by some students.

Question 2

Just under forty percent of students attained full marks in this question, which required completion of a trace table in accordance with a pseudo-code algorithm. The most common error here was entering the correct value in the wrong column, which might be due to errors of transposition rather than any misconceptions.

Question 3

This question required students to describe a difference between a global and a local variable, and fewer than half of students succeeded in providing a description that secured both marks. Although there was an understanding that the scope of a local variable is more limited, this was often phrased too vaguely to be fully creditworthy.

Question 4

This question required a definition of the term ‘algorithm’, and most students failed to pick up either of the two marks. The most common misconception was that an algorithm is ‘a set of instructions’. Since there is no semantic difference between ‘a set of instructions’ and ‘an instruction set’, credit could not be given for such answers.

Question 5

Converting pseudocode to program code was carried out successfully by most students, with approximately ninety percent of students picking up either full marks or eight out of nine on question 5.1. The most common way in which a mark was dropped was reversing the logic of the selection statement.

Where the code was sound, the evidence mark of 5.2 was almost always awarded. Students are reminded of the importance of ensuring that screenshot evidence is legible in all such questions.

Question 6

For each of the four question parts, most students gained the mark. Here, students were required to identify a variable and three subroutines. The most common way in which marks could not be awarded was by the inclusion of additional code, such as parentheses alongside subroutine identifiers.

Question 7

For both question parts, most students gained the mark. As this question related to data structures rather than subroutines, additional code was less frequently included here.

Question 8

This question required students to identify abstraction as the underlying principle in a given situation, and eighty percent of students were able to do so.

Question 9

Here, students were required to demonstrate an understanding of the skeleton code by identifying error codes that would be generated under certain circumstances. Seventy percent of students picked up either full marks or four out of five.

Question 10

This question continued to examine students' understanding of the error codes within the skeleton program, and the majority were awarded both marks on offer. In circumstances where one mark was obtained but two points had been made, there was often a re-phrasing of the first point in an attempt to pick up the second mark.

Question 11

Question 11 comprised three parts, each of which asked about some aspect of the same subroutine. In previous exams, this might have been presented as a single six mark question rather than a series of related three-mark questions, and in such questions the scale of the required answer presented a challenge to some students.

This format provided some structure, and most student marks were concentrated between two and five out of six. Where marks were not awarded, it was quite common for an otherwise creditworthy answer to be placed next to the wrong question. This suggests that marks were lost due to a misunderstanding of the question rather than a misunderstanding of the code.

Question 12

This question was the first that required changes to be made to the skeleton program, and answers were of a good quality, with around seventy percent of students securing either full marks or three out of four on question 12.1.

Where a mark was dropped in an otherwise correct response, it was that awarded for updating the status register, which was missed out rather than being attempted incorrectly.

Question 13

This question required addition of validation code; the requirement was that any specified line number should exist within the assembly program. Although the lower bound was generally correct, since it would always be 1, the upper bound was commonly hard-coded rather than determined using a variable. Since the length of the assembly program could vary, hard-coding was not enough for the mark.

Approximately half of the students secured either full marks or four out of five for question 13.1, although very few students were awarded two or three marks. That indicates that this question marks the beginning of the stage in which individual programming marks were cherry-picked due to constraints of time or understanding.

Question 14

This question was skipped more than any other question, possibly due to students focusing on the higher-scoring question 15.

Of those who made an attempt, three quarters secured either full marks or three out of four on question 14.1. A range of different approaches was taken, including some not anticipated by the mark scheme. These were always given credit, some obtaining full marks, and this underscores the fact that there are usually many correct answers to a programming question, not just one.

Question 15

There were two components to this question – adding the facility to insert an instruction and the facility to remove one. As is often the case with the last question on the paper, there were many partial attempts, and this question lent itself to such attempts. Some students attempted to write code for insertion only, and others for deletion only. Only a minority of those who made any attempt at all were awarded zero marks for question 15.1, with students securing marks for modifying the menu and handling the new inputs even if they did not complete the task.

Questions 15.2 and 15.3 required screenshot evidence for a successful deletion and insertion respectively. Given that a large number of students had only attempted one of these two pieces of functionality, a high number of non-attempts were registered. Nevertheless, the majority of students who made a partial attempt at 15.1 also attempted the corresponding screenshot evidence question.

Mark Ranges and Award of Grades

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