AQA

A level
Computer Science

Paper 2
Report on the Examination

7517/2
June 2017

Version: v1.0

1.1
Most students demonstrated a good understanding of the aspects of the fetch-execute cycle covered by this question part, with approximately half scoring three or four marks. Common misconceptions were that the address bus was internal to the processor and connected the PC and MAR and that registers are capable of carrying out activities and not just memory stores; for example, a response might include a phrase such as “the MBR sends”. The phrasing of some responses also suggested that the registers themselves moved rather than the contents of them, for example “the MBR moves along the data bus”.

1.2
This question part was intended to test students’ understanding of the fact that the MBR is used during the execute phase of the fetch-execute cycle so its contents could be overwritten but this was not grasped by many students. Some students gave good responses relating to parallel execution, which were considered creditworthy within the simplified model of a processor that is used by the specification. As with part 1.1, some students associated abilities with registers that they do not have, for example writing that “the MBR cannot decode”, suggesting that they believed that the CIR could.

1.3
Most students appeared to know the fundamental differences between the Harvard and von Neumann architectures and many responded to this question by writing about these differences rather than describing advantages of the Harvard architecture. Good responses recognised that instructions and data could be fetched simultaneously, resulting in the processor spending less time waiting for memory fetches. This potential speed increase was the focus of most mark worthy answers, although some also touched on points about security and different word lengths for data and instructions. Common misconceptions were that Harvard would avoid collisions between data and instructions, and that for individual machine code instructions the opcodes would be stored in instruction memory and the operands in data memory.

2.1
This question was very well answered; approximately 90% of students were able to perform the decryption. The most common error was to encrypt the already encrypted data for a second time instead of decrypting it.

2.2
This question was well answered, with three quarters of students achieving some marks. Students who lost marks generally did so because their answers were too vague or unclear rather than because they were fundamentally wrong. For example, a response might state that “only the sender knows the key” when clearly the recipient would need to know it as well even if it was kept securely, or did not make clear that a point that was being made related to the key by, for example, writing that “it must be totally random”.

2.3
Just over half of students correctly explained that symmetric encryption used one key for both encryption and decryption whereas asymmetric used different keys for each process. Some responses lost marks through lack of clarity, for example by stating that asymmetric used two different keys without indicating that one was used for encryption and the other for decryption. A small number of students confused the topic with asynchronous and synchronous communication.

3.1
This question was not well answered, with only about a third of students achieving the mark. These students correctly recognised that with eight different transmission levels the value of three bits could be sent in one group as $2^3 = 8$. Common incorrect responses were 8, 4 and 2^8.

3.2
This question part was better answered than 3.1. Many students who answered 3.1 incorrectly still recognised that they needed to multiply their answer to 3.1 by 500 for this part and follow-through marks were available for students who answered 3.1 incorrectly but performed the correct calculation in this part. The most common error was to simply multiply 500 by 8.

3.3
This question part was well answered with just over two thirds of students correctly recognising that there was a linear relationship between bandwidth and bit rate and that the correct answer was B. The second most commonly selected response was A.

3.4
Two thirds of students showed a good understanding of this topic and were able to achieve some marks for explaining the advantages of serial communication over long distances. This was a question where knowing the correct terminology was helpful, for example using the phrase “data skew” provided a much easier route to getting a mark than trying to describe what data skew is as many descriptions did not clearly show that the student understood the concept. For example, just stating that “all the bits might not arrive together” did not clearly convey that a student understood data skew. Some statements were not considered mark worthy as they are true of serial communication too, which also suffers from interference and corruption. Responses stating that serial communication was cheaper required further explanation of why this was the case to achieve marks.

4.1
This question part was well tackled with over three quarters of students drawing a fully correct logic circuit. If students make a mistake whilst drawing a logic circuit and they cannot correct it in a clear way then they are advised to redraw it on a new sheet of paper as it can be difficult to discern what type of gate the student had drawn if, for example, an AND gate is drawn on top of an OR gate.

4.2
This question part was well tackled. Students need to be aware of the significance of brackets as the expression (A.B).(C+D) is not logically equivalent to A.B.C+D as the AND (.) operation has a higher order of precedence than OR (+). Three quarters of students achieved both available marks.

4.3
Most students achieved some marks for this question part but only a fifth achieved full marks. The most commonly made mistake was to incorrectly apply the identity $\mathrm{A} + \overline{\mathrm{A}} = 1$ to the subexpression $\overline{\mathrm{A}} + \mathrm{A} \cdot (\mathrm{A} + \mathrm{B})$ which failed to recognise that this could not be done because of the order of precedence of AND (.) and OR (+). Another mistake made by some students was to cancel NOTs when they could not be cancelled, for example believing that the NOTs in $(\overline{\mathrm{B}} \cdot \overline{\mathrm{C}})$ could be cancelled with the longer NOT that related to the entire expression.

4.4
Good responses recognised that a D-type flip-flop can be used as a form of memory to store a state or the value of a bit. Some students talked themselves out of achieving a mark by stating that a flip-flop would retain its state when the power to the circuit is turned off, which is incorrect.

4.5
This question was not well answered. Many students stated correctly that the other input would be a clock or trigger signal, but explanations of the purpose of this were less good. Mark worthy responses recognised that input on the clock signal caused the flip-flop to store the state of its input and output this state until the next input on the clock signal was received. Commonly seen responses that were not mark worthy were that the clock caused the state or the output of the flip-flop to change, both of which failed to recognise that the change of state/output was to reflect the current data input. A small number of students describe the internal logic of the flip-flop; knowledge of this is not required for the specification. A further small group of students appeared to confused D-type and SR flip-flops, stating that a D-type flip-flop would have set and reset inputs.

5.1
Two thirds of students were able to explain that the operand contained a value that would be operated upon by the opcode. Fewer successfully explained that the addressing mode would be used to determine how the operand should be interpreted (in this specification) as either a value to be used directly or as the address of the memory location that stored the value to use. Students needed to relate the addressing mode to the operand to get the mark; it was not sufficient to state that the addressing mode indicated if direct or immediate addressing was to be used.

5.2
Almost all students seemed to be familiar with the AQA assembly language instruction set and were able to achieve some marks for this question, with approximately two thirds achieving at least three of the four available marks. The vast majority successfully loaded the value 10 into R1 to achieve the first mark. The second mark was for correctly performing the shift of the value in R1 and storing the result in R2. Some students went wrong at this stage by shifting incorrectly, for example by performing the shift in decimal rather than binary or by shifting in the wrong direction. The third mark was for adding the two values in R1 and R2 together and storing the result in R1 and also loading the value 50 from memory location 101 into R3. This mark was the second most frequently awarded mark for this question. The final mark was for storing the value 1 into both R4 and memory location 102. Many students did the former but not the latter so failed to achieve this mark.

5.3
Just under a fifth of students were able to correctly identify the purpose of the program which checked if the value in memory location 101 was five times the value in memory location 100. This was inevitably a hard mark to achieve as it relied on a student having correctly completed the trace in 5.2 and then going on to analyse the overall purpose of the program.

5.4
Students were more familiar with advantages of high-level languages than disadvantages of them. Most commonly students wrote about high-level languages having easier to understand code, facilitating faster development times and being more portable. Some answers also focussed on facilities that would be available in high-level languages such as significant libraries of built-in functions and data structures. Some responses were too vague to be mark worthy, for example stating that program code written in high-level languages was “close to English”.

Good responses about the disadvantages of high-level languages recognised that a compiler might generate machine code that was less than optimal in the sense of using more memory than required or executing more slowly. Other good points related to limitations such as the inability to control directly the contents of registers or specific memory locations. Points related to different translation methods such as compilation taking longer than assembling were not considered mark worthy.

6.1
This question was moderately well answered with just over half of students correctly identifying that the result of applying the functions was 10. Some students gave the result as [10], which was accepted this year. Students should be aware however that, when the output of the head function should be a single value, we will not accept answers expressed as lists in the future. The most common incorrect response was 15.

6.2
The purposes of the `map` and `filter` functions were fairly well understood with two thirds of students achieving two or more of the three marks, but the purpose of the `fold` function was less well known. The most common incorrect response with regard to filter was [10, 15] which suggested that students had either confused < and > or that they believed the `filter` function filtered numbers out of the result rather than filtering them into it. `fold` was poorly understood. Many students either gave the original list as the output or simply appended the 0 to the original list to generate the output. Marks were only awarded for `fold` if the result was expressed as a single value rather than a list containing a single value as the fundamental purpose of this application of the fold operation was to combine the list into one single value.

6.3
The term “higher-order” function was not well understood with only just over a quarter of students correctly identifying that a higher-order function would take another function as an argument or return a function as a result. Responses that a higher-order function would use other functions were not considered mark worthy as this description could be applied to many functions. Other common mistakes were to believe that a higher-order function was simply more important than some other functions or to confuse one with a built-in function in a procedural language.

7
This question required students to write an extended response that covered a number of different areas of the specification. Responses often covered the aspects of how the owners could have protected their networks and what legal and ethical issues might have arisen well but neglected the aspect of how it was possible for the data to be collected and were not specific enough with regard to how the company’s practices might have changed as a result of the incident.
With regard to how the data was collected, examiners were looking for students to recognise that WiFi signals could travel over a wide area and that any WiFi receiver in range could read the data from these packets. Few students made these points, but a reasonable number identified that the data may not have been encrypted or that an outdated encryption protocol such as WEP might have been used.

Good responses recognised that an appropriate measure to prevent the data from being collected would have included encrypting the data using a protocol such as WPA2. In the question’s context, measures such as enforcing a MAC address whitelist or using a firewall were not appropriate as the cars were simply collecting information that was being transmitted; they were not trying to connect to the wireless access points. Some students referred to adding a password to the network which was not a strong enough point to be mark worthy; they failed to identify that the "password" they were referring to would actually be used to make a key for an encryption system. Students are not required to have knowledge of specific legislation for this specification, but should have an understanding of issues around areas of ethical and legal concern such as privacy, data protection, copyright and hacking. Nevertheless, mention of specific relevant legislation was considered mark worthy. Relatively few students took the opportunity to really discuss the legal and ethical issues. For example: given that the data was being transmitted freely through the air, would accessing it really count as hacking, given that the functionality involved was added by a small number of developers, were they or the company responsible for it, would the nature of any offences committed depend upon the type of data collected, did it matter if the data collection was intentional or accidental? Students should be encouraged to consider this sort of reflection when answering this type of question.

Most students recognised that the company needed to have better oversight of the development process but many made general statements about this and did not suggest specific measures that might be taken to improve their practices, such as introducing third-party review of code or improved training of developers on legal and ethical issues. Points relating to testing were not considered mark worthy unless it was explicitly stated that the testing would be focussed on ensuring that the software had no additional functionality or that the data collected by the cars was all relevant to the intended purpose of the system.

Some students appeared to have completely misunderstood the scenario and wrote at considerable length about the rights and wrongs of companies using cars to take photographs in public places rather than about interception of WiFi signals. Students need to ensure that they are answering the question asked rather than a question that they might have prepared for.

8.1
This question was fairly well answered with most of students achieving full marks for correctly calculating the size of the sample. Even those students who did not achieve both marks usually achieved one for some correct working, with the most common error being failing to convert a response to kilobytes correctly for example by never converting from bits to bytes. This year we accepted responses that equated 1024 bytes to 1 kilobyte, but students should note that the specification defines a kilobyte as being equivalent to 1000 bytes. In the future, we will not accept responses that use 1024 and, in any case, working is likely to be more complex if 1024 is used.

8.2
The vast majority of students recognised that this question related to Nyquist's theorem and were able to successfully state what this was. Some students recognised that double something was involved, or even double a frequency but were not able to offer sufficient clarity to be awarded a mark whilst good responses clearly recognised that the sample rate must be at least twice the frequency of the highest frequency component in the signal being sampled. To achieve the second mark, students needed to use the data given in the question and recognise that 20,000 was less than double 14,500 so to satisfy Nyquist's theorem a minimum sample rate of 29,000 Hz should have been used.

8.3
Most students achieved some marks for this question but only a fifth scored three or four marks. The advantages of MIDI over sampled sound were better known than how data was represented. For the latter, students generally recognised that properties of notes such as pitch were stored but did not explain that these would be stored as a sequence of event messages, or instructions that would comprise the entire piece of music.

The most commonly stated advantages of MIDI representation were that it produced smaller file sizes and that it was more easily editable. However, these advantages whilst often stated were not always well understood, with some students believing incorrectly that the smaller file size resulted from MIDI taking fewer samples. Students also often held the incorrect belief that it was not possible to edit sampled sound at all and did not appreciate that MIDI simply gave greater control over the editing of individual notes than sampled sound did, as a result of the representation identifying these. Some students stated that MIDI would produce a better quality sound. Whether this is true or not depends upon a number of factors but responses that explained that it avoided errors that may be introduced during the sampling process for sampled sound were accepted.

9.1
More students were correctly able to identify a correct IP address for port A and port B. The most common error for part A was to give an IP address which started with the octets 187.17.10 which were the octets on the other port of the same router. Students need to be aware that IP addresses cannot have a value of all zeros or all ones (binary format) in the host ID part. Having all bits as zero or all bits as one in the host ID of an IP address is not allowed as these values have special purposes.

9.2
More students successfully described a physical topology than a logical topology. This is unsurprising given that the latter is a somewhat abstract concept. Good descriptions of a physical topology recognised that this referred to the layout of the cabling that connected the devices together. Students often missed out on this mark by just stating that the physical topology related to how devices were connected or wired together. This description did not capture the concept of layout and could equally have referred to, for example, the choice of the type of cable used. Good descriptions of logical topology identified that the logical topology would define how the data flowed through the network, for example the protocol that was used. A relatively common misconception was that a physical topology was a wired network and a logical topology was a wireless network.

9.3
This topic was one of the least well understood topics that were covered on the question paper. Just a quarter of students were able to achieve three or four marks. Students often either (1) confused server-based with thick client and peer-to-peer with thin client or (2) confused server-based with a star physical topology and peer-to-peer with a bus physical topology. Many students referred to all of the processing being done on the server, each device having its own connection to the server, all communication going through the server or “everything on the network” being stored on the server, none of which are true.

Good responses recognised that some resources (eg files, web pages, databases) would be stored on the server in a client-server system and that the server would provide these resources to clients in response to requests from them. They went on to identify that in a peer-to-peer system resources would be stored locally on each computer and these computers could share their resources with any other computer on the network.

Many students wrote about peer-to-peer systems in the context of BitTorrent file sharing systems. This was acceptable but students should be aware that this is just one type of peer-to-peer system and that it is not the case that peer-to-peer systems must be connected to the Internet or that in every peer-to-peer system files are split into parts and distributed across multiple computers. Another common misconception was that client-server systems had to be connected to the Internet.

10.1
Approximately a third of students correctly identified that a composite key made up of `CarRegNo` and `JobDate` could have been used. The most common incorrect response was `CarRegNo` on its own. This could not have been used as it would not have allowed a car to be booked into the garage for more than one job.

10.2
Good responses recognised that a customer might own more than one car so by separating the owners into a new table data redundancy and the possibility of data inconsistency would be reduced. Where students have been given a scenario in the question it is important that they make reference to this when responding. Responses such as "it would make the database more normalised" were not considered to be enough to be mark worthy as the student had not shown that they understood why the original design was not normalised nor that they appreciated the benefits of making it normalised.

10.3
Three quarters of students gained some marks for completing the entity-relationship diagram. Some students used non-standard notations which were not considered creditworthy or drew more than three relationships on to the diagram. Students who draw more relationships than are asked for in this type of question face having their maximum mark limited.

10.4
Whilst three quarters of students achieved a mark in this question part for correctly identifying the data that would be required to perform this update, only a third achieved marks for producing syntactically correct SQL. Common errors were to use `INSERT INTO` or `SELECT` instead of `UPDATE`, to specify the fieldname in the `UPDATE` clause instead of the table name and to miss out the condition necessary to identify which record to update.

10.5
This was the least well-answered of the question parts that required students to write SQL code. More students believed that they should have used an `UPDATE` query than realised that they needed to use `INSERT`. Of those who did use `INSERT`, the most common errors were to miss out the keyword `INTO`, to miss out brackets where they were required or to put the numeric quantity value into quotation marks.

10.6
This was the best-answered of the SQL question parts. The majority of students knew the correct structure of query to search for data although there was considerable variability in the accuracy of their solutions and just under a fifth of students gained full marks. Common syntactical errors were to add in brackets where they were not required, eg into the list of fieldnames or around `ASC`, to put commas or semi-colons between each clause or to use the keyword `AND` in the `SELECT` and `FROM` clauses in place of a comma. When a query requires that data from more than one table is used, students need to remember to include an appropriate condition to join the data from the two tables (either using an `ON` clause or additional `WHERE` conditions). A small minority of students attempted to write two distinct queries to retrieve the required data separately from each table.

10.7
Just under half of students achieved some marks for this question. A common incorrect response was to add the `Make` and `Model` attributes to the `Part` relation. This was not an appropriate solution as it would have only allowed each part to be fitted to a single make/model of car, without introducing data duplication in the parts table. Students who identified the appropriate solution of

creating a new relation containing the attributes `PartID`, `Make` and `Model` often achieved two marks rather than three as they did not identify the appropriate primary key for the table.

11.1
This question was well answered with just over two thirds of students correctly identifying that C was a normalised negative value. The key to doing this was identifying that the digit to the left of the binary point was a 1 and the digit to the right was a 0.

11.2
Just under two thirds of student recognised that B was the smallest positive normalised value. This was most quickly achieved by spotting that the value was the only positive normalised value as it had a 0 to the left of the binary point and a 1 to the right of it.

11.3
This question was fairly well tackled and approximately half of students achieved both marks. Those who did not usually achieved one mark for shifting the binary point by the correct number of places but then made a calculation error when converting to decimal.

11.4
This question was very well tackled, with two thirds of students achieving full marks. The most common mistakes were to shift the binary point by five places instead of six or to give the exponent as minus six instead of six.

11.5
This question was well tackled with over three quarters of students achieving the mark. Students need to be aware that the absolute error is always expressed as a positive number; the most common error was to give the answer as -0.05 instead of 0.05.

11.6
Just over two thirds of students could correctly calculate the relative error. Of the students who did make errors, the most common one was to divide 0.05 by 13.75, ie the value that actually was represented instead of 13.8 which was the original value to be stored.

Mark Ranges and Award of Grades

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