COMPUTING

Paper 9691/11
Written Paper 1

General comments

The standard of candidates' work was broadly similar to previous years.

The format of the examination was similar to June 2013 and the candidates seemed better prepared for this new style of paper than they were six months ago. The new format leads candidates and Centres gradually into the revised syllabus which has its first sitting in June 2015.

Candidates need to understand that it is no longer the case that simply learning sections from text books will score many marks on future papers. There is a need to understand how to apply their knowledge in new scenarios (both in examinations and in their learning of the subject). This will hopefully give a better grounding of the subject but also enable candidates of all abilities to perform to their full potential in the examination system.

Comments on specific questions

Question 1

This question was well answered with the full range of marks being shown. The majority of candidates chose observation, interviewing and questionnaires as the best methods for fact finding.

Question 2 (a)

Candidates needed to develop their understanding of this question as low marks were achieved. In general, marks were awarded for use of colours (red = off) and suitable input devices (such as a touch screen). The use of graphics or how graphics could be used in the monitoring process was not mentioned in many answers.

Question 2 (b)

This question was generally well answered with touch screen and keyboard being the most commonly chosen input devices. A significant number of candidates also gave some good reasons for their choice of device.

Question 3 (a)

This question was generally well answered; however, candidates need to improve their understanding of verification and validation in order to gain full marks.

Question 3 (b)

A number of candidates confused the two parts of the question and gave range check in part **(i)** and length check in part **(ii)**; however, the majority of candidates correctly identified the validation checks in both parts.

Question 3 (c)

In part **(i)**, a large proportion of the candidates correctly calculated the check digit to be 4. However, a significant number ignored the note in the stem of the question which indicated that the check digit was in position 1. Those that missed or ignored this statement did the calculation incorrectly and returned a check digit of 2. Fortunately, many candidates showed all their working and were usually credited with some marks even though the final answer was incorrect.

Part **(ii)** was reasonably well answered with many candidates indicating that the check digit for 34921 would be 7 and not 1. This would indicate an error in transmission.

Question 4 (a)

This question was answered well. However, candidates would have benefited from further understanding as:

- many candidates drew AND and OR gates with only 1 input which lost them most of the marks
- some of the drawings were not very good and it was very difficult to work out which logic gate was being drawn; fortunately, many candidates wrote the name of the gate inside each symbol which allowed Examiners to give them credit.

Essentially candidates need to be more careful when drawing the logic symbols to make sure that marks are not lost due to lack of attention to detail.

Question 4 (b)

This was generally well answered with many candidates gaining 3 or 4 marks. A large number of candidates had used the *working area* in the table to make sure that their final answers were correct.

Question 5 (a)

This part of the question posed no real problems. Although a significant number of candidates were still writing "it is a message sent to the processor …" which is incorrect.

Question 5 (b)

There were some reasonable attempts at answering this question. The most common error was to suggest that it was the processor that tried to do the editing of the document whilst the printing was done in the background. Very few candidates understood that the user wanted to carry on editing one document whilst they printed out something else.

Question 5 (c)

This question was reasonably well answered, with a large number of candidates understanding how virus checkers worked in principle.

Question 5 (d)

A large number of candidates gained one mark for "allows one user to use the computer at a time"; however, very few achieved the second mark for reference to multi-tasking or use of profiles to ensure security via user ids/passwords. Marks were lost for writing "allows one user to use the computer" since this omits the key phrase "at a time".

Question 6 (a)

This question was generally well answered but very few candidates gained the full three marks. A large number of candidates did not realise that "RAM is volatile while ROM is non-volatile" is actually only one difference and not two. The question required three distinct differences. Many candidates also did not know that volatile, temporary and contents of memory lost when switching off power were all in fact the same thing.

Question 6 (b)

This question was generally well answered with the full range of marks seen.

Question 6 (c)

Candidates needed to show better understanding of this question. Since sensors are usually part of any data logging system, naming a sensor was not acceptable as a method of automatic data capture. Answers such as keyboards, touch screens and mice chosen as suitable devices were unclear.

Question 6 (d)

Candidates needed to improve their understanding of sensors. Many candidates believe that sensors only take readings when something has happened or changed. They did not seem to realise that sensors usually take readings constantly and are "sampled" at a given time interval by a data logger or computer/microprocessor. Whilst there are sensors which only produce data when something happens (e.g. breaking a light beam) this would not be the case in this scenario.

Question 7 (a)

The majority of candidates obtained half marks or less. Candidates needed to improve their knowledge of the main features of bus, star and ring network topologies.

Question 7 (b)

This part was well answered with many candidates gaining the full marks.

Question 8 (a)

As expected, most candidates knew at least two features of spreadsheet software; but there were many imprecise responses such as "they do calculations" or "you can make graphs" – a little more expansion is needed to gain the marks e.g. *spreadsheets allow automatic recalculation to be done when the data in one or more cells is changed*.

Question 8 (b)

Candidates needed to show better understanding of this question. Some imprecise answers were seen, such as "more expensive", "take time to get it" or "need training" with no explanations given.

Question 8 (c)

This question was well answered with many candidates choosing industrial applications such as chemical factories or nuclear energy plants.

Question 9 (a)

Candidates needed to improve on their answers for this question. There were some imprecise responses such as "they act as an expert"; many marks were lost because of unclear answers of this type.

Question 9 (b)

Candidates needed to further develop their answers to this question. Many candidates were fully aware of rules base, knowledge base and inference engines but lacked understanding about how they all interact. Candidates also needed to develop their knowledge of input/output interfaces.

COMPUTING

Paper 9691/12

Written Paper

General comments

The standard of candidates' work was broadly similar to previous years.

The format of the examination was similar to June 2013 and the candidates seemed better prepared for this new style of paper than they were six months ago. The new format leads candidates and Centres gradually into the revised syllabus which has its first sitting in June 2015.

Candidates need to understand that it is no longer the case that simply learning sections from text books will score many marks on future papers. There is a need to understand how to apply their knowledge in new scenarios (both in examinations and in their learning of the subject). This will hopefully give a better grounding of the subject but also enable candidates of all abilities to perform to their full potential in the examination system.

Comments on specific questions

Question 1 (a)

Candidates needed to further develop their knowledge and understanding of this question. Many candidates thought that OMR did the checking against the answer sheet/template to provide the final mark. In general, the question was not very well answered with few references to how the shaded lozenge was identified and read and how the coordinates of the lozenges gave their position on the answer sheet – this allowed it to be compared with the stored template.

Question 1 (b)

In part **(i)**, a large number of candidates understood that compression reduces the size of the file. There were two common errors that lost many candidates marks:

- "compression saves space" – space is always regarded as too vague and never gains any credit
- "reduces size of data" – the size of data is not reduced, it is the file size that is decreased.

The next part was well answered with many correct references to MP3/4 files, jpeg and zip/rar files.

Question 2

This question was very well answered with the full range of marks seen from 0 to 10. The most common error was to link only one item on the right to one item on the left. This error meant that at least 6 marks were lost by the candidate. It is essential that candidates read the question very carefully to ensure they know exactly what is required to gain the full marks. The question stated: "*Draw a line connecting each description to the correct component*" – it was made clear in the previous paragraph that the components were shown on the left and the statement was very clear that EACH description had to be connected to a component.

Question 3 (a)

This question was well answered. A common error was to see "full duplex is to send data in both directions" which is not enough for a mark since it is essential to state "at the same time/simultaneously".

Question 3 (b)

There were no real issues with this question; the concept of protocols seems to be well understood.

Question 4 (a)

Whilst about half of the candidates correctly identified CAD as the software for drawing the rides, it was very common to see imprecise answers, such as *design software*, or incorrect answers, such as *desk top publishing* or *presentation software*. However, those that correctly identified CAD also did well in answering the part about features of the software.

Question 4 (b)

Candidates needed to develop their understanding of this question. Many candidates thought the question was referring to intruders or people behaving badly in the leisure park. They claimed that the cameras were used solely for security reasons to watch visitors to the park.

Very few candidates seemed to understand the phrase in the question "… to maximise the safety of the rides". This referred to things like not overloading the rides (e.g. use of pressure sensors) or tracking their speed (again with sensors). Nor did candidates realise that cameras could take images of the rides to see if there were any breakages, missing bolts or even corrosion (e.g. rust) – this could be done by comparing these new images with those already stored in memory. This was a case of reading the question carefully and deciding exactly what was needed to gain a good mark.

Question 4 (c)

Candidates needed to develop their understanding of this question. Many candidates gained 1 or 2 marks for reference to flashing red images warning of danger or the use of touch screens. Very few candidates made any reference to the use of graphics to depict the movement of the rides or how the three screens could be used in the monitoring process. This was again a case of reading the question carefully and deciding exactly what was needed to gain a good mark.

Question 5 (a)

This question was answered well with no real problems to report.

Question 5 (b)

Many candidates ignored the last part of the question and lost marks for not including a diagram as clearly requested.

Question 6 (a)

Candidates needed to develop their answer to this question. Many candidates copied the input values into the output columns or simply applied the NOT function to all the input values. Also a number of answers appeared to be random. Better candidates gained 3 or 4 marks here once they understood the description of the scenario in the question.

Question 6 (b)

There were three possible answers here depending on the interpretation of how the three outputs R, G and Y could be generated. There were many good answers with full marks obtained in a number of cases. However, some of the drawings of the logic gates were so unclear that it was impossible for Examiners to work out which gate it was meant to be. In many cases, candidates wrote the name of the gate inside the symbol which made it possible to award marks. Candidates need to take more care when drawing logic gates since many marks were lost by lack of attention to detail.

Question 7 (a)

This question was generally well answered; but marks were lost for the following reasons:

- the statement in the question asking for each validation check to be different was ignored by many candidates
- candidates gave some very imprecise, even unknown, validation checks e.g. "check it does not contain a decimal point".

Question 7 (b)

In part **(i)**, about half the candidates realised this was a consistency check and that “Mr” and “Male” need to be chosen together in the *title* field and *sex* field to pass the validation check.

In part **(ii)**, a large number of candidates only mentioned format checks on the date and missed the fact that it was necessary to check that the date they joined the company must be a later date than their birth date.

Question 8 (a)

This question was well answered with many candidates gaining full marks. Where marks were lost, it was due to very imprecise descriptions in the second part of the candidates answer in each case.

Question 8 (b)

Both parts were well answered with no problems to report.

Question 9 (a)

Candidates needed to improve their knowledge for this question. The majority thought it was a user interface/HCI or thought it was the expert system.

Question 9 (b)

Candidates needed to develop their knowledge for this question. Most appeared to know the terms used (rules base, inference engine or knowledge base, for example) but did not seem to know how all the components of an expert system linked to each other to make it work.

Question 9 (c)

The majority of candidates gained 1 or 2 marks here for “they lack common sense”, “users need to undergo much training” or “they need to be frequently updated to work”. There were several references to costs but these were too imprecise to be given any credit.

COMPUTING

Paper 9691/13
Written Paper

General comments

The standard of candidates' work was broadly similar to previous years.

The format of the examination was similar to June 2013 and the candidates seemed better prepared for this new style of paper than they were six months ago. The new format leads candidates and Centres gradually into the revised syllabus which has its first sitting in June 2015.

Candidates need to understand that it is no longer the case that simply learning sections from text books will score many marks on future papers. There is a need to understand how to apply their knowledge in new scenarios (both in examinations and in their learning of the subject). This will hopefully give a better grounding of the subject but also enable candidates of all abilities to perform to their full potential in the examination system.

Comments on specific questions

Question 1 (a)/(b)

Both parts of this question posed no problems for the majority of candidates. The basic definitions seem to well known.

Question 2 (a)

Again parts **(i)** and **(ii)** were very well answered by the majority of candidates.

Question 2 (b)

This question was generally awarded 1 mark for most candidates. Very few gained the second mark for a satisfactory expansion of their answer. Essentially most candidates correctly identified that the character set corresponded to the symbols shown on a standard computer keyboard.

Question 2 (c)

A large number of candidates made reference to the use of ASCII codes; but only the better candidates also indicated that characters are represented by a unique binary code in the computer.

Question 3 (a)

Whilst about half of the candidates correctly identified CAD as the software for drawing the rides, it was very common to see imprecise answers, such as *design software*, or incorrect answers, such as *desk top publishing* or *presentation software*. However, those that correctly identified CAD also did well in answering the part about features of the software.

Question 3 (b)

Candidates needed to develop their knowledge and understanding of sensors. Many thought that the sensors "decide" if the water is too hot or too cold. They also suggested that the sensors send signals to the valves or microprocessor to open or close the valves to regulate the water flow and temperature. This misconception about the function of sensors continues to cause problems with a very significant number of candidates. It is the microprocessor/computer that makes the "decisions" – sensors merely convey the data and do not carry out any actions.

Question 4 (a)

Candidates needed to further develop their understanding for this question. Many gave examples of hardware where media type was required; for example, DVD, when the answer should be optical (media). This lost candidates many marks. Also candidates should be aware that CD-R and CD-RW, for example, are regarded as only type of device; similarly for DVD. So again marks were lost here for this type of error.

Question 4 (b)

Candidates needed to improve their knowledge of this topic area. The full range of marks was seen in this question. Marks were usually lost for very general or imprecise answers where the candidate did not appear to be too sure about what happened. For example, answers such as "data is sent to the buffer" are far too imprecise since it does not say where the data comes from or which buffer it is sent to. At this level, candidates need to be much clearer in their descriptions.

Question 5 (a)

Again the full range of marks was seen here. Most marks were lost where candidates gave items found in technical documentation rather than in user documentation; for example, flowcharts and program listings.

Question 5 (b)

Candidates needed to improve their understanding of this question. The only marks awarded were for a reference to colour and the type of interface. Many candidates thought the interface should appeal to children and gave features that would appeal to a 5 or 6 year old. It is very unlikely that the shop terminals would cater for that option.

Question 6 (a)

A full range of marks was seen in this question. As on all three papers this year, there is some confusion about the features of bus, star and ring topologies. The errors in drawing the connecting lines showed no real obvious pattern of where candidates were going wrong. It might help candidates when revising this topic to produce their own diagrams (similar to the one on the exam paper); this could help them understand which features correspond to each of the network topologies.

Question 6 (b)

Candidates needed to improve their understanding of this question. Very few candidates gained more than half marks here. They were aware of packets moving independently and the need to reassemble the packets at the destination. However, very few indicated how each packet is identified or how this method compared to conventional switching. Answers were too general with little real understanding of the full mechanism of packet switching.

Question 7 (a)

Candidates needed to develop their understanding of this question. There were long discussions about off-the-shelf software. The question asked for a discussion of the impact of software, in general, on the work place. For example, potential unemployment, need for training, possible working from home and increased productivity.

Question 7 (b)

This question was answered well. In some cases, marks were lost for simply giving one of the drawbacks as the opposite of an already-given benefit. For example: benefit (off-the-shelf) – less expensive since development costs can be shared by many users; drawback (custom-written) – more expensive since only one user pays for all the costs. These are just the same point with the argument switched. Candidates need to be aware that to gain both marks, different benefits and drawbacks need to be given.

Question 8 (a)

This question was answered well with many candidates gaining full marks.

Question 8 (b)

This question was answered well. However:

- many candidates drew AND and OR gates with only 1 input which lost them most of the marks
- some of the drawings were unclear and it was very difficult to work out which logic gate was being drawn; fortunately, many candidates wrote the name of the gate inside each symbol which allowed Examiners to give them credit.

Essentially candidates need to be more careful when drawing the logic symbols to make sure that marks are not lost due to lack of attention to detail.

Question 9 (a)

There were no problems in this question. Many candidates gained the full marks.

Question 9 (b)

There were no problems in this question. Many candidates gained the full marks.

COMPUTING

Paper 9691/21
Written Paper

Key message

For this component candidates need to have practical programming experience in the programming facilities of a chosen high-level language (such as Basic, Visual Basic or Pascal). Candidates need to write their own programs, adapt programs written by others and write programs from pseudocode prepared by others. They should also have practical experience of using debugging tools as specified in the syllabus.

Question 1

(a) The better candidates gave answers which included points such as: modularisation allows debugging of a small section at a time, makes the program easier to maintain and produces reusable code. Weaker answers did not distinguish between easy to debug and easier to debug. Candidates need to understand the difference. A significant number of candidates thought that more than one programmer could be employed here. Candidates need to appreciate the scenario given in the question and not rely too much on answers seen in mark schemes of past examination papers.

(b) Candidates needed to improve their understanding for this question. There was evidence that many candidates had not come across structured diagrams in any form. Many described and then drew a variety of data flow type diagrams in the second part.

(c) Candidates needed to improve their understanding for this question. Many different ways of passing data values were described. Most candidates seemed to think that it just happened.

(d) Either parameter passing or global variables were acceptable here.

(e) Candidates needed to improve their knowledge for this question. Their ideas on how sequential files worked were diverse. Many thought that a data item in an array could not be accessed directly, or that data had to be entered in sequence. They did not appear to appreciate the role of an array index.

(f) The question stated that a two-dimensional array was required. Candidates, who clearly had practical experience of declaring and initialising arrays, could give very precise answers.

Question 2

(a) This question was generally answered well, particularly part **(iii)**.

(b) Candidates had a good try at this question and generally set it out well. They gained marks for the first condition, nesting and indenting, but often got the wrong conditions to express the 2nd and 3rd ranges of marks. Only a few were able to use Modulo 2 to find the odd and even numbers.

(c) Candidates needed to improve on their answers for this question. A variety of languages were proposed, but few candidates knew the correct CASE construct for their language. Fewer still were able to express the ranges needed within their language's individual cases.

Question 3

(a) Many purposeful screen layouts were seen. The better candidates clearly distinguished between those data items that the user is required to enter – and therefore text boxes are needed – and those data items the program will display to the user – and therefore labels are needed.

(b) This was answered nearly as well as part **(a)**, but many candidates did not know how to group the information in the report.

(c) This question was answered well.

(d) Only the better candidates appreciated how much change was needed in this problem to go from a FOR loop to a REPEAT..UNTIL loop. Many candidates used a REPEAT and then a FOR loop inside that. They gained few marks.

(e) (i) Most candidates could respond to some of this question.

(ii) Many intelligent guesses were written in response to this question. Generally a break point was where a program could be broken and stepping points were places you could step over. Overall very few candidates had actually used these tools for debugging.

Question 4

Recursion is a challenging topic. The better candidates clearly understood the concept and could answer this entire question correctly. Most candidates answered part **(b)** correctly, and many did well in their answer to part **(c)**.

COMPUTING

Paper 9691/22
Written Paper

Key message

For this component candidates need to have practical programming experience in the programming facilities of a chosen high-level language (such as Basic, Visual Basic or Pascal). Candidates need to write their own programs, adapt programs written by others and write programs from pseudocode prepared by others. They should also have practical experience of using debugging tools as specified in the syllabus.

Question 1

(a) Many creditworthy answers were seen to this question. Candidates need to remember that a structure diagram shows the modules at a particular level in the order in which they would be executed. Each module box of the diagram should contain the module description starting with a verb. For example, the second level modules should be: input employee's data, calculate the bonus, calculate deductions, print out the bonus.

(b) Most candidates could list program coding techniques. Fewer candidates could describe why program code written using these techniques would be helpful during program maintenance. Creditworthy answers included: comments to help understand a section of code; indentation to show structure; meaningful variable names to convey the purpose of the variables; capitalised identifiers and/or capital keywords to make these words stand out.

(c) (i) Candidates needed to further develop their understanding of this question. Very few creditworthy answers were seen. Candidates often stated that using a local variable would avoid possible conflict. Only the better candidates could adequately explain that local variables have scope only in the module they are DECLARED in.

(ii) The better candidates could explain that the use of parameters would allow passing values from one module to another. Candidates need to understand that global variables are not a satisfactory method to make values available to other modules as the self-containment of modules is lost.

(d) Most candidates made a good attempt at this question. Candidates need to take note of the question. There were plenty of possible choices: Normal data for the other job types, boundary data, both just within or just outside the boundary, and invalid data.

Question 2

The question stated that a two-dimensional array was required. Candidates, who clearly had practical experience of declaring and initialising arrays, could give very precise answers.

(a) Candidates needed to further develop their understanding for this question. There was evidence that many candidates were unsure how to declare the dimensions of a two-dimensional array. This varies with different languages and candidates are expected to have worked with arrays using the programming language of their choice.

(b) (i) The better candidates knew that initialising an array is good programming practice because it ensures that only known values are stored. Some candidates are under the wrong impression that initialisation reserves the memory locations for the array. Candidates need to understand that in most programming languages variable and array declarations reserve the memory locations.

(ii) Candidates need to understand that initialising every element of a two-dimensional array requires two nested loops. Their control variables (for example i and j) are used as the index variables for the initialisation, for example: MyEbooks(i,j) = "".

(c) Very few candidates realised that the program needs to write the data stored in the array out to a file before the program terminates. Many candidates gave answers that were too imprecise to earn credit. When the program is run again the file needs to be opened and the data read back into the array.

(d) Most candidates correctly stated the data types and field sizes for the given identifiers.

Question 3

(a) Many purposeful screen layouts were seen. The better candidates clearly distinguished between those data items that the user is required to enter – and therefore text boxes are needed – and those data items the program will display to the user – and therefore labels are needed.

(b) Most candidates made a good attempt at completing the trace tables. A frequent omission was the number of attempts in the first column. Candidates need to work very carefully through the algorithm and not jump to early conclusions. For question part **(ii)**, many candidates wrote an output in the final column when in fact no output would be produced for the given input.

(c) Most candidates correctly stated that the algorithm contained a logic error. The better candidates were able to rewrite the required line. There are several correct answers, for example: UNTIL (Password = "Aisha") OR (Attempt = 3).

(d) Most candidates correctly stated that someone's name is too easy to guess to be a good password.

(e) Most candidates described methods of choosing a more secure password, such as using a mixture of upper and lower case letters as well as digits. Fewer candidates suggested that a longer password would be appropriate. The better candidates also mentioned that personal data that is widely known, such as names or telephone numbers are not suitable passwords.

(f) (i) Candidates needed to develop their understanding of this question. The candidates who clearly had programming experience gave largely correct answers. Many candidates did not appear to know how to nest IF statements or how to call the appropriate procedure depending on the user's choice. Candidates should be reminded that they need to use the variable identifier given in the question: Choice.

(ii) Candidates need to have practical experience of using the CASE/SELECT statement to really understand how it works in their chosen programming language.

Question 4

(a) Recursion is a challenging topic. The better candidates clearly understood the concept and could trace the given calls perfectly. Weaker candidates appeared unaware of the fact that the function called itself in line 6.

(b) The better candidates correctly stated that the purpose of line 1 was to give a terminating condition. Candidates, who really knew the topic of recursion, stated that this was the base case. Candidates should understand that a recursive subroutine must consist of a general case and a base case and that the base case must be reached after a finite number of recursive calls.

(c) Very few candidates could explain that, if x is less than y, the first recursive call reverses the values of x and y and then the function carries on normally.

COMPUTING

Paper 9691/23
Written Paper

Key message

For this component candidates need to have practical programming experience in the programming facilities of a chosen high-level language (such as Basic, Visual Basic or Pascal). Candidates need to write their own programs, adapt programs written by others and write programs from pseudocode prepared by others.

Question 1

(a) Most candidates could answer the first two parts of this question, but the responses to the third part were often imprecise.

(b) The better candidates gave answers which included points such as: modularisation allows debugging of a small section at a time, makes the program easier to maintain and produces reusable code. Weaker answers did not distinguish between easy to debug and easier to debug. Candidates need to understand the difference. A significant number of candidates thought that more than one programmer could be employed here. Candidates need to appreciate the scenario given in the question and not rely too much on answers seen in mark schemes of past examination papers.

(c) Only a few candidates could give specific reasons as an answer to the first part. While most candidates could draw the second part, many gave some form of data flow diagram, and carried this on to part **(d)**.

(e) The better candidates were able to work out that the only data value mentioned was the one that was passed between the two modules. In part **(ii)** the better candidates could explain that the use of parameters would allow passing values from one module to another. Candidates need to understand that global variables are not a satisfactory method to make values available to other modules as the self-containment of modules is lost.

Question 2

(a) This question was generally well answered.

(b) This question was also well answered, but a number of candidates had boxes 1 and 2, or boxes 5 and 6 in the wrong places.

(c) Candidates needed to improve on their answer for this question. For a simple basic piece of code this was answered with a great many mistakes, even in the declaration of an array.

(d) Many candidates had a good try at this question. Several wrote in more of a story document than pseudocode. Typical errors involved not getting the correct conditions to produce the given ranges and not knowing how to sort out those values ending in a zero. Many candidates did not nest their IF statements.

(e) The candidates who had used a CASE statement during the year answered this well. The majority though, did not know how a case statement was used in their chosen language, or how to deal with a range within each case.

Question 3

(a) This question was well answered with most of the main requirements entered.

(b) This part was often well answered, but many candidates did not seem to know of any way of structuring the report.

(c) This question was answered well.

(d) Candidates were asked to use the techniques mentioned in part **(a)**. Most used indentation, but none of the others. The better candidates did not use a FOR loop within a REPEAT loop.

(e) Candidates needed to improve their knowledge of definitions for this question. Although these terms are not easy to define, they are common terms and little understanding was shown.

Question 4

This question was answered well.

Many candidates fully understood the recursive process involved and produced good answers.

COMPUTING

Paper 9691/31
Written Paper

General

Computing is a practical subject. Candidates are required to demonstrate their practical programming skills on paper 2. Candidates should also not lose sight of the fact that many of the topics on the paper 3 syllabus have a practical bias. It is appreciated that some topics, such as networking, may be difficult for a Centre to teach in a practical way as they are heavily dependant on resources. However, evidence on this paper showed weak responses, especially for the question on Data Manipulation Language. There is a wealth of free software available for the study of SQL. It can only be assumed that the weaker answers seen are an indication that candidates are not being exposed to practical work in this area. This conclusion was also a possible explanation for the weak answers seen for **Question 4** on object-oriented programming.

Question 1:

Some candidates provided correct answers to parts **(a)(i)**, **a(ii)** and **c(ii)**. In part **c(ii)**, the inclusion of an '=' in the expression seemed to provide an extra problem. Many candidates correctly answered part **(b)**. Most seemed able to convert the reverse Polish expression to infix 'manually' with a small minority using a stack to demonstrate how they arrived at the answer. In part **(c)(i)** and part **c(iii)**, few correctly named the appropriate traversal method; descriptions could have been clearer, and 'node' rather than 'root' was indicative of weak understanding. Answers expected were either the terms 'in order traversal and 'post order traversal', or a description of the order – 'left-root-right' and 'left-right-root'.

Question 2:

Candidates needed to improve on their knowledge for this question, as in part **(a),** three correct advantages were very rare. The most common, correct, answers were those relating to data duplication and data inconsistency. Answers that mentioned "saving space" or "easier searching" were common but did not gain credit. Despite the information given in the question, answers to part **(b)** were often incorrect with a significant number of candidates giving "one-to-one" as their answer. Candidates should appreciate that the order in which the entities are stated in the question is significant; an answer of 'many-to-one' was correct but 'one-to-many' was not.

Creditworthy answers to part **(c)** depended upon a correct answer to part **(b)(i)**; consequently there were few answers that showed a link table. Where a link table was given, problems with an appropriate name for the table and also its relationship with Runner and Race ensured very few answers received full marks.

Candidates needed to develop their answers for part **(d)**. Candidates need a firm grasp of the practical application of 1NF, 2NF and 3NF. They need to further develop their knowledge of SQL queries, as in part **(f)** it was evident that candidates had little practical experience of writing SQL queries. Many answers did not use the information supplied; typically only one of the fields appeared in the SELECT part of the query. The SELECT, FROM And WHERE parts were often in the wrong order and the condition was often lacking an AND operator.

Question 3:

Most candidates showed some knowledge of the Von Neumann architecture and many were able to make two valid points.

In part **(b)**, those candidates who managed part **(i)** correctly could correctly answer the conversion in part **(ii)**. Some answers demonstrated that candidates needed to develop their knowledge and understanding of the hexadecimal number system. Candidates needed to improve on their answers for part **(iii)**, with *"easier to*

understand" and *"takes fewer bytes / less space"* common. Many answers incorrectly suggested hexadecimal numbers took up less storage space.

Answers to part **(c)(i)** needed further development as candidates were unable to realise that a 4 digit hexadecimal number would occupy 2 bytes. Most candidates scored some marks on part **(c)(ii)** but very few achieved full marks. The part of the table associated with the buses was found to be more challenging.

Part **(d)** was one of the best answered questions on the paper with all candidates able to score.

Question 4:

Candidates needed to improve their knowledge for part **(a)**, as they often scored one mark, but two and three were rare. In part **(b)**, the answers seen suggested that candidates needed more practical experience of object-oriented design and programming. This was evident from the class diagrams presented. Good answers were rare; common problems included: no attempt to show parent and child relationships with the given classes; repeating attributes in child classes; no attempt to show inheritance; introducing attributes not mentioned in the stem of the question and the omission of a data type for attributes.

Candidates needed to develop their knowledge and understanding for part **(c)**, as a significant minority of candidates offered no answer. Only a small minority of answers showed any understanding of encapsulation.

Question 5:

This was the best answered question on the paper. The operation of a stack, (last in – first out) was identified by most candidates in part **(a)**. In part **(b)(i)** many gave the correct values. Attempts at the missing pseudocode in parts **(b)(ii)** and **(c)** needed further development in order to achieve full marks. In the PushJob procedure, adding the new user ID to the array caused more problems than any other addition. In PopJob, identifying a value that indicated that the stack was empty seemed to be the most challenging of the three additions. Many candidates understood in part **(d)** that the stack was unsatisfactory because of how it operated and that a queue (using first in – first out) was a more appropriate choice for this application.

Question 6:

Candidates needed to develop their knowledge and understanding of a FAT. Many answers were based on the idea that the file allocation table is used to allocate / locate files and confused the FAT with the file directory. There were also many answers that seemed to be referring to the storage of files in memory. There was greater understanding of the underlying theory in part **(b)(i)** but creditworthy answers were rare.

For part **(b)(i)**, the use of a stack in step one was unknown but the other three gaps did demonstrate that some candidates knew about ISRs and that the contents of registers saved need later to be reloaded. Answers to part **(b)(ii)** needed greater understanding; many candidates being distracted by 'priority' and consequently writing about priority queues and job priorities.

Question 7:

In part **(a)**, 'email' and 'passwords' were a good source of marks for candidates. A common weakness was for candidates to give applications which were imprecise: "military", "hospitals", or "security application". The reasons for the encryption were also often not clear.

In part **(b)** many answers were typically " *…an algorithm that encrypts the message …".* Candidates found it difficult to describe algorithm and encryption in their own words. The encryption key was, for many candidates, what is used (only) to decrypt a message. Part **(c)** provided further evidence that many candidates did not understand the concept of keys and, in particular, private and public keys. Many answers referred to a public key as if there was only one public key. Also, few candidates showed any understanding that private keys have an owner. Candidates found **(d)** more accessible and full marks were not unusual. User IDs was the common answer for authorisation; authentication attracted all three methods given on the mark scheme.

COMPUTING

Paper 9691/32
Written Paper

General

Computing is a practical subject. Candidates are required to demonstrate their practical programming skills on paper 2. Candidates should also not lose sight of the fact that many of the topics on paper 3 have a practical bias. It is appreciated that some topics, such as networking, may be difficult for a Centre to teach in a practical way as they are heavily dependent on resources. However, evidence on this paper showed weak responses, especially for the question on Data Manipulation Language. There is a wealth of free software available for the study of SQL. It can only be assumed that the weaker answers seen are an indication that candidates are not being exposed to practical work in this area. Candidates need to have practical experience of using SQL for writing queries, writing SQL commands to add a record, delete a record and amend a record in a table.

Question 1:

Some candidates provided correct answers to parts **(a)(i)**, **(a)(ii)** and **c(ii)**. Part **(a)(i)** was better answered than the other two question parts. A majority of candidates correctly answered part **(b)**. Most converted the reverse Polish expression to infix 'manually'. A small number of candidates used a stack for demonstrating how they arrived at their answer. Candidates would have benefited from better understanding of question parts **(c)(i)** and **c(iii)** as few candidates correctly named the appropriate traversal method. The mark in each case could be gained either with a statement of the method in-order or post-order traversal, or by stating the order in which the nodes were visited – left-root-right and left-right-root respectively. Descriptions could have been clearer, and 'node' rather than 'root' was indicative of weak understanding.

Question 2:

Candidates needed to improve on their knowledge for this question, as in part **(a),** three correct advantages were very rare. The most common, correct, answers were those relating to data duplication and data inconsistency. Answers that mentioned "saving space" or "easier searching" were common but did not gain credit. Despite the information given in the question, answers to part **(b)** were often incorrect. Many candidates gave one-to-many as their answer to part **(b)(i)**. Candidates should appreciate that the order in which the tables are given in the question is significant; an answer of 'many-to-one' is correct but 'one-to-many' is not. Creditworthy answers to part **(c)** depended upon a correct answer to part **(b)(ii)**. Only a small number of candidates successfully drew a diagram that showed an appropriately named link table and the correct relationships with PRODUCT and ORDER.

See the general comments made above about the need to expose candidates to practical tasks on areas of the syllabus. Candidates needed to improve their knowledge for part **(d)**. There were many contradictory answers such as: "The table is not in 3NF because there are no dependent non-key attributes" or "The table is in 3NF because …". Many answers started with a 'Yes' or 'No' perhaps indicating that the candidate had not read the question carefully enough. Answers seen suggest this is a topic which needs further understanding. This is an example of a question where the ability to reproduce a bookwork definition of each of the Normal Forms will not gain credit. Candidates must be able to apply this knowledge to a given practical table design. In part **(iv)**, many candidates lost marks because they did not underline the primary key. Many candidates did not include the foreign key SupplierID in the PRODUCT table. Any marks awarded were usually for a correct SUPPLIER table. Again in part**(f)**, it was evident that candidates had little practical experience of writing SQL queries. Many answers did not use the information supplied (typically only one of the fields appeared in the SELECT part of the query).

Question 3:

Candidates needed to further develop their understanding of registers; answers were often just a list of registers (Index register, etc.).

Candidates needed to improve their knowledge of this topic area. In part **(b)** answers were often in binary or showed an inability to convert 12 into the hexadecimal character 'C'. A common error in part **(b)(ii)** was to treat the '10' as ten and return an incorrect answer of 175. In part **(b)(iii)** common misconceptions were: a hexadecimal equivalent to a binary number would take up less memory space; hexadecimal numbers have a greater range. The answer expected was simply that a hexadecimal representation will be written using less digits (and so be less prone to error) compared to binary.

In part **(c)(i)** "4 bytes" was a common, incorrect answer. Only a few candidates gave "2 bytes". Many answers to part **(ii)** showed evidence of some understanding of the F-E cycle and the use of the various registers. A description of how the contents of the MDR are derived proved to be the step that caused most difficulty. This has been commented on in a previous Report and does require good communication. The key statement is that the instruction copied to MDR is *"the data value found at the address given in MAR".*

In part **(d)** only a small number of candidates displayed evidence that they could interpret successfully the assembly language instructions.

Question 4:

Candidates needed to improve their understanding for part **(a)**. The relationship between classes and objects was often understood to be one where a class contained many objects. In part **(b)** a class diagram that exhibited the correct subclasses and the correct inheritance was rare. However, most candidates scored some marks – usually for the correct distribution of attributes amongst the classes. Many diagrams did not show the relationships between the classes and many did not recognise which items of data belonged to which class. Also an attribute was given but the data type was not included. The same attribute being present in more than one class was a common mistake.

There was a little more understanding shown of encapsulation than that of classes and objects. Data hiding / access through methods defined in the class tended to be the more popular correct answers. This is again a topic where candidates should be exposed to practical work on basic object-oriented programming and design.

Question 5:

In part **(a)**, the majority of candidates identified 'Boolean' correctly. In the algorithm, the incrementing of the index and the final test were the best source of marks. The majority of candidates preferred, incorrectly, 'CustomerName' to 'SearchName'. In part **(b)**, 2 or 100 were more likely to be given than 50. This was intended to be an accessible mark with the candidate appreciating that a sequential search of a list of 100 items would require 50 comparisons, i.e. the item will be found - 'on average', the key words in the question stem - half way through the list. A few candidates understood that the items needed to be in order in part**(c)(i)**. More candidates could indicate how the give function displayed recursion. A mark for the first '6' was common in part **(iii)**. Correct values of Low and High thereafter proved difficult for most candidates. There was much confusion over what the function returned and when it happened. The return of a name was a common misunderstanding.

Question 6:

Candidates performed better in part **(a)** on addition. Most correct comments came from giving 'overflow' as the answer. Many answers in part **(b)** were incorrect because they treated the 12 bits as a fixed point fraction. Correct answers tended to use the movement of the binary point as the method to produce the right number. Knowledge of how to recognise normalised numbers was often successfully stated. The consequences of changing the number of bits allocated to the mantissa and exponent was also well understood.

Question 7:

In part **(a)**, 'email' and 'passwords' were a good source of marks for candidates. A common weakness was for candidates to give applications which were imprecise: "military", "hospitals", or "security application". The reasons for the encryption were also often not clear.

Candidates needed to improve their knowledge for part **(b)** as it was clear that many had not made the connection between the terms and encryption. Consequently the answers were often about different types of text: text which only had alphabetic characters, only letters, text that had hyperlinks, images added to it, or composed of symbols. The vast majority of answers missed the ideas of text before encryption and text after encryption.

In part **(c)**, despite the information given, private and public keys were mentioned by a number of candidates. Many candidates rewrote the question as their response: "the message is encrypted using a single key …".

Candidates found **(d)** more accessible and full marks were not unusual. User IDs was the common answer for authorisation; authentication attracted all three methods given on the mark scheme.

COMPUTING

Paper 9691/33
Written Paper

General

Computing is a practical subject. Candidates are required to demonstrate their practical programming skills on paper 2. Candidates should also not lose sight of the fact that many of the topics on paper 3 have a practical bias. It is appreciated that some topics, such as networking, may be difficult for a Centre to teach in a practical way as they are heavily dependent on resources. However, evidence on this paper showed weak responses, especially for the question on Data Manipulation Language. There is a wealth of free software available for the study of SQL. It can only be assumed that the weaker answers seen are an indication that candidates are not being exposed to practical work in this area. Candidates need to have practical experience of using SQL for writing queries, writing SQL commands to add a record, delete a record and amend a record in a table.

Question 1:

Some candidates provided correct answers to parts **(a)(i)**, **(a)(ii)** and **c(ii)**. In part **c(ii)** the inclusion of an '=' seemed to provide an extra problem. Many candidates correctly answered part **(b)**. Most seemed able to convert the reverse Polish expression often by first converting the expression to infix. A small number of candidates used a stack for evaluating the expression. Candidates would have benefited from better understanding of question parts **(c)(i)** and **c(iii)** as few candidates correctly named the appropriate traversal method. Descriptions could have been clearer, and 'node' rather than 'root' was indicative of weak understanding.

Question 2:

In part **(a),** three correct advantages were very rare. The most common, correct, answers were those relating to data duplication and data inconsistency. Answers that mentioned "saving space" or "easier searching" were common, but did not gain credit. Despite the information given in the question, answers to part **(b)** were often incorrect with a significant number of candidates giving "one-to-one" as their answer. There were many answers which did not appreciate that a normalised solution would be achieved by using a link table. Where a link table was given, problems with an appropriate name for the table and also its relationships with CAR and CUSTOMER ensured very few answers received full marks. Where a link table was used, a common error was to show the two one-to-many relationship lines the wrong way around.

For part **(d)** the rubric of the question tried to lead the candidate, by stating for parts **(i)** and **(ii)** that the table was not in 1NF and 2NF respectively. Some candidates clearly had learnt a format definition of each of the three normal forms, but were unable to apply and use this knowledge with the particular table design given.

Candidates needed to develop their knowledge of SQL queries, as in part **(f)** it was evident that candidates had little, if any, practical experience of writing SQL queries. Many answers did not use the information supplied; typically only one of the fields appeared in the SELECT part of the query.

Question 3:

Few candidates understood what was meant by a register. Answers expected were that it is a temporary storage location found inside the microprocessor. Some candidates did understand there were registers which did a specific role such as the Program Counter and others, whereas others were general purpose registers which temporarily stored data read from the main memory. Candidates need to make the connection here between a 'register' and the 'Accumulator (ACC)' which was referred to in part **(d)**.

Candidates needed to improve on their answers to parts **(b)(i)** and **(ii)**. Some candidates often gave a binary string for a question which asked for a hexadecimal answer.

For part **(b)(iii)** a simple statement such as *"it requires less digits in hexadecimal to represent a number"* would have secured the mark. However, this does not mean that hexadecimal will take up less space in memory which was a common incorrect answer.

Most candidates scored for part **(c)(ii)** with statements describing the sequential steps in the fetch stage or by writing the equivalent register transfer notation.

Question 4:

Candidates scored well on this question for parts **(a)** and **(b)** appreciating that some goals will return a true/false value rather than one or more data values from the set of facts.

Part **(c)** was well answered by the stronger candidates who appreciated the need for a common variable within the three relevant clauses and their connection with two AND operators.

Part **(d)** was less well answered with candidates asked to explain the backtracking required to evaluate the given goal.

Question 5:

All candidates were able to score on this question. Most understood that the pseudocode would require a test for reaching the 'end of the file' and that the file should finally be 'closed'.

Part **(b)** was intended as a straightforward mark, but this proved not to be so. For a sequential search 'on average' (the key words in the question stem) the search will take 125 comparisons - that is the required value will be found half way through the list.

The question framework for part **(c)(iii)** was new and candidates clearly understood the way – using the embedded boxes – that the recursion was being represented.

Question 6:

Most candidates were able to score for their explanation of the boot file. The most popular answer was that it has information about where the operating system is to be found and will trigger the loading of the operating system software.

Answers for part **(b)** were generally strong with candidates able to give a formal definition of an interrupt and then follow this with examples of interrupts which are either software or hardware generated.

Many candidates were able to secure the full three marks for part **(c)**.

Question 7:

For part **(a)**, this was a new style question framework which was used to examine the communications section of the syllabus. Candidates were clearly comfortable in answering this style of question, but answers were weak. Very few scripts were seen which secured the full six marks.

Answers seen for part **(b)(i)** had improved; as this had been examined before and the Examiner's Report stated that many candidates did not appreciate the term 'media' in the context of computer communications. A wide range of answers were all considered appropriate for part **(b)(i)** including various forms of copper wiring, optical fibre cable and one of the many forms of radio wave communications.

COMPUTING

Paper 9691/04

Project 2

General comments

This report provides general feedback on the overall quality of project work for GCE Advanced Level Computing candidates. In addition, all Centres receive specific feedback from their Moderator in the form of a short report that is returned after moderation. This reporting provides an ongoing dialogue with Centres giving valuable pointers to the perceived strengths and weaknesses of the projects moderated.

The projects submitted covered a wide variety of topics with better candidates showing evidence of researching a problem beyond their School or college life.

In order to have the full range of marks available to the candidate, the computing project must involve a third party client whose requirements are considered and clearly documented at all stages of the system development. Centres are reminded that the project work is designed to test the candidates' understanding of the systems life cycle. The requirements are clearly set out in syllabus ***section 4***, 'The Guidance on Marking the Computing Project' ***section 7.2*** acts as a useful checklist, for teachers and candidates, setting out the expected contents of each section.

Centres are reminded that this guidance and the marking scheme changed in 2011. Please use the up-to-date A Level Computing Syllabus for guidance on project choice, content required and how to assess candidates' project work.

Centres are strongly reminded that candidates should use this guidance for the expected contents of their reports rather than some of the A Level textbooks available for project work, which do not cover the full requirements of the CIE syllabus. Candidates who prepare their work only using these text books and not the syllabus for guidance, may miss out vital sections of their reports; or complete unnecessary work, for example, feasibility studies and cost benefit analysis.

Project Reports and Presentation

As usual, the presentation of most of the reports was to a very high standard, with reports word-processed and properly bound. Candidates should ensure that only material essential to the report is included so that they only submit one volume of work. Candidates are reminded that only authentic letters from clients and/or users must be used to provide evidence for the Evaluation, Implementation, Investigation and Analysis sections. These letters could be scanned in to the project report, but must not be re-typed/typed out by the candidates.

It is strongly recommended that the structure of the candidate's report follows that of the mark scheme set out in the current syllabus. Many candidates are structuring the report based on the mark scheme for the previous syllabus with different/extra sections and the inclusion of evidence no longer required. Essential evidence should not be relegated to appendices. This allows both teachers at the Centres, and Moderators to easily check that work for all sections has been included. Also, it is essential that the pages of the report are clearly numbered by the candidate.

Project assessment and marking

Nearly all Centres used the marking grid in the current syllabus to provide a breakdown of marks showing the marks given for each sub-section of the report. However, in order to aid the process of moderation, the completed grid should include references to the appropriate pages in the candidates' reports where evidence for each section can be found. Also, teachers should comment as to why they awarded the marks for each

section. Moderators have noticed that where there is a good commentary provided by a teacher, the marking is usually very close to the agreed standard.

Section 3 Comments on Individual Sections

The comments set out below identify areas where candidates' work is to be praised or areas of concern, and are not a guide to the required contents of each section.

(a) Quality of report.

Most candidates set out their reports in the appropriate sections and made good use of illustrations including diagrams and screenshots. Weaker candidates sometimes did not include page numbers in their reports, this meant that teachers could not clearly identify to the Moderator where evidence was to be found and those candidates were unable to cross reference items within their report.

(b) Definition Investigation and Analysis

(i) Definition - nature of the problem

Most candidates described the organisation and many identified the methods used, but only the better candidates described the methods used and also described the origin of the data and indicated the form of the data. This is a brief introduction for anyone who is unfamiliar with the organisation and the area under investigation.

(ii) Investigation and Analysis

In order to gain good marks, candidates must clearly document client and user involvement and clearly state agreed outcomes. Candidates need to consider carefully the evidence obtained from interviews, observation of the existing system and study of documents currently in use; then ask follow up questions to fill in any gaps in the knowledge obtained about the current system or requirements needed for the new system. Alternative approaches need to be discussed in depth as they would be applied to the candidate's proposed system. A detailed requirements specification should be produced based on the information collected, this must include the specific requirements of the system to be produced and not just concentrate on hardware and software.

(**c) Design**

(i) Nature of the solution

The requirements specification set out in the analysis needs to be discussed with the client and a set of measurable objectives agreed. These objectives will then form the basis for the project evaluation. Most candidates provided designs that included proposed data structures, layouts for input screens and reports required; better candidates used pseudocode and/or flowcharts to provide a detailed description of the processes to be implemented.

In order to obtain good marks for this sub-section, candidates need to obtain evidence that their client has seen and commented on the design work, and then show what has changed as a result of these comments. Many candidates do not complete this step and thus cannot be awarded top marks for this sub-section. Evidence from the solution is not required here.

(ii) Intended benefits

In order to obtain good marks for this sub-section, candidates should describe the benefits of their intended system, not just provide a list of general statements that could apply to any system.

(iii) Limits of the scope of solution

Candidates should describe the limitations of their intended system including an estimate of the size of any files required, not just provide a list of general statements that could apply to any system.

Full marks for the design section cannot be awarded without candidates clearly supplying evidence for **(i)**, **(ii)** and **(iii)**.

(d) Software Development, Programming Testing and Installation

(i) Development

Evidence of development should include program listings of code written by the candidate, data structures used and evidence of tailoring of software packages. To achieve higher marks, this should be error free, match the design specification in **(c)(i)** and be annotated by the candidate.

(ii) Programming

It is important that the programming code in this sub-section is written by the candidate and not produced as a result of tailoring a software package. Marks should only be awarded to code that has been written by the candidate.

Candidates need to show that they can apply the programming skills developed at AS level in Paper 2 to a real situation. This includes technical programming competence and ensuring that their program could be maintained by writing self-documented code.

(iii) Testing

Evidence of testing needs to be supported by a well-designed test plan that includes the identification of appropriate test data, including valid, invalid and extreme cases, together with expected results for all tests. To achieve high marks, the test plan should clearly identify that all parts of the system have been tested. Many candidates only tested the validation and navigation aspects of their system, and omitted to test that their system did what it is supposed to do, for example, production of reports. This omission meant candidates were unable to gain marks in the highest band for this sub-section.

(iv) Installation

Most candidates provided an implementation plan containing details of user testing, user training and system changeover.

For good marks to be awarded written evidence from the client and/or user(s) must be included in order to show that the system has been seen, used and tested, and the candidate's plans have been agreed.

Centres are reminded that appropriateness of structure and exploitation of available facilities have not been required for this sub-section of the report since 2011.

(e) Documentation

(i) Systems Maintenance Documentation

This sub-section of the report is a Systems Maintenance document. Many candidates incorrectly included Technical Documentation. Please see the current syllabus for details of what should be included in this sub-section.

For top marks to be awarded, the candidate must explain how adaptive maintenance could be undertaken for their system.

(ii) User Guide

This section was completed to a good standard by most candidates. Centres are reminded that for full marks the candidate must include an index and a glossary for their User Guide. This needs to be complete including details of how to install the new system, backup routines and a guide to common errors. Also, good on-screen help should exist where this is a sensible option.

(f) Evaluation

Centres are reminded that in order to gain high marks, candidates need to provide a detailed evaluation that includes the content set out in the guidance for marking projects section of the

syllabus. Many candidates provided limited evidence for this section; if this is the case then there are few marks that can be awarded.

(i) Discussion of the degree of success in meeting the original objectives

Candidates need to consider each objective set out in **(c)(i)** and explain how their project work met the objective or explain why the objective was not met.

Candidates should also indicate where the evidence, probably from testing or feedback from the users of the system, could be found in their report to support these conclusions.

(ii) Evaluate the client's and users' response to the system

A response must be provided directly from the client and user(s) showing that they have used the system, not just reported by the candidate. The candidate should then evaluate their client's and users' responses.

For evidence in this section to be creditworthy, the candidate must include original letters, preferably on headed notepaper, signed by the client and not typed and/or composed by the candidate.

Centres are reminded that possible extensions and the good and bad points of their final system are not required for this sub-section of the report.