GCE

Computing

Advanced GCE **A2 H447**

Advanced Subsidiary GCE **AS H047**

Reports on the Units

June 2010

HX47/R/10

Oxford Cambridge and RSA Examinations

OCR (Oxford Cambridge and RSA) is a leading UK awarding body, providing a wide range of qualifications to meet the needs of pupils of all ages and abilities. OCR qualifications include AS/A Levels, Diplomas, GCSEs, OCR Nationals, Functional Skills, Key Skills, Entry Level qualifications, NVQs and vocational qualifications in areas such as IT, business, languages, teaching/training, administration and secretarial skills.

It is also responsible for developing new specifications to meet national requirements and the needs of students and teachers. OCR is a not-for-profit organisation; any surplus made is invested back into the establishment to help towards the development of qualifications and support which keep pace with the changing needs of today's society.

This report on the Examination provides information on the performance of candidates which it is hoped will be useful to teachers in their preparation of candidates for future examinations. It is intended to be constructive and informative and to promote better understanding of the specification content, of the operation of the scheme of assessment and of the application of assessment criteria.

Reports should be read in conjunction with the published question papers and mark schemes for the Examination.

OCR will not enter into any discussion or correspondence in connection with this report.

Any enquiries about publications should be addressed to:

OCR Publications
PO Box 5050
Annesley
NOTTINGHAM
NG15 0DL

Telephone: 0870 770 6622
Facsimile: 01223 552610
E-mail: publications@ocr.org.uk

CONTENTS

Advanced GCE Computing (H447)

Advanced Subsidiary GCE Computing (H047)

REPORTS ON THE UNITS

Chief Examiner Report

General comments

This was the first full session of the qualification H447. The standard of work from the candidates was pleasing, particularly in the project work with some of the work of a very high standard.

The papers worked largely as intended and the examiners were happy to report that the form of the questions and the ways in which the questions were asked did not have a detrimental effect on the performance of the candidates. This was despite the fact that many of the questions needed to be asked within a context which the candidates are certain to have different amounts of knowledge of. This can cause a problem for candidates who know more about a topic than is required by a question because they tend to read extra information into the question than is provided by the stem. It is useful when answering scenario based questions to ensure that the knowledge used is contained within the body of the question and not to add extra information of which the candidate may be aware. This is particularly important in F452 where the questions must be asked in context. In these cases all the necessary information will be available in the question, if a candidate relies on facts which they happen to know then any additional information may not be worthy of credit.

The Principal Examiner's report for F453 highlights the poor use of technical terms by many candidates, who should be able to use basic terminology like storage and memory correctly. There is a need for improvement in basic examination technique. Candidates should be aware that there is a correlation between the number of marks available for a question and the space offered for the response. The number of marks available for the question are clearly stated at the end of the question and candidates should ensure that they have given enough information to allow the examiner to award the marks. If a question is worth three marks and the candidate only makes two points then they cannot normally earn full marks, likewise if the command word is explain, then a list of points will be insufficient.

F451 Computer Fundamentals

General comments

The paper seemed to work well and discriminate right across the mark range. Some questions were testing for even the most able candidates while there were still many questions where the weaker candidates could accrue marks.

There were some excellent scripts received and the candidates responsible can be proud of their efforts in what was neither intended to be an easy paper nor did it prove to be. The presentation of the responses was generally of a high standard and again candidates can be congratulated for the way their responses were presented. Standards of written English were particularly impressive with great improvements being made in this area, however, there are still occasions where deciphering the meaning was sometimes reduced to spotting the use of some keywords. Candidates need to be able to express themselves if they are going to expect to earn the marks that their computing knowledge justifies. Many candidates found the use of bullet points in many of the questions a sensible way of planning their response. While this would be inappropriate to the type of response expected for 4a it is a sensible approach to many of the other questions.

There did not appear to be any indication of general misunderstanding of the questions and there was no evidence of any candidates getting into trouble with the time allocation. The use of the paper to write the answers on seemed reasonable with there being few examples of candidates running out of space or of offering continuations to responses on additional sheets. The final mark scheme is published by the Board and the attention of centres is directed toward it for a comprehensive listing of appropriate answers to each question

Comments on individual questions

Question 1

This was a good starter question, no candidates failing to score at least some marks and a good distribution of marks towards the top end of the marks.

(a) Most candidates scored well. A minority failed to identify sensible devices the worst example being the storage device but the majority managed to score full marks.

(b) A little bit more difficult because the answers required were not actually from the candidates' experience unlike part (a) but the majority once again managed to score well here.

(c) i was well answered but ii caused more problems with marks almost equally distributed across the full range.

(d) The calculation was well answered with most candidates successfully showing their method by making the carry values clear. The parity question on the other hand proved more of a challenge. Most candidates managed to use parity as their reasoning and picked the correct byte and then gave justification to varying degrees of success. However there was a large minority who looked for any pattern that they could find. Many chose the first byte because it ended with a zero. A significant group made it far more complicated choosing the last byte because it was the only one with the last two bits the same. Could this have been a confusion with normalising floating point numbers?

Question 2

An excellent discriminator which produced an almost perfect bell shaped distribution of marks. While accepting that some candidates do not recognise the significance of the keyword 'describe' in the question and hence limit themselves to a maximum of 4 marks out of the 8, it was still disappointing to see such a relatively small proportion of candidates able to describe four purposes of an operating system. This sort of answer is often typified by the type of response: 'It controls the hardware so that the hardware is controlled'. Such basic failure of simple examination technique is something that many centres might find useful to concentrate on when studying the past paper questions in future.

Question 3

A well answered question with most candidates scoring well on both parts a and bi. Bii proved more challenging but the more able candidates were still able to earn both marks.

Question 4

(a) A very good discriminator. Candidates were expected to talk about both parts of the question to decide the band of marks that the response should be placed into and then the quality of the evidence offered was used to decide which mark point they should get within that band. Responses ranged from simple lists of fact finding methods to comprehensive discussions about the need to find out the information requirements and then the full explanation of a number of methods.

(b) This should have been a simple part of the question because many candidates would have had practical experience of completing these items of documentation for projects which they may have completed or for which they are preparing for later in this course. These system specifications are not specifically Computing in nature, although some of the items that would have attracted marks are Computing linked.

Question 5

Some good quality and complete responses here. Problems were caused for candidates who decided that the word octal in the question should be interpreted as hexadecimal. This is not intended as a cynical or facetious comment because it is important to those candidates who made the mistake which almost certainly arose because at the front of their mind was the hexadecimal example that they did as revision on the last of the past papers just before the exam. This is a problem that teachers need to guard against. Yes, it is important that candidates practise on materials before the exam but we must be careful that they don't assume too much from such practice. The other common errors here were a confusion between two's complement and sign/magnitude which meant that the 6 marks in part b were not accessible and the failure of most candidates to point out that the final carry out of the byte in the calculation was discarded.

Question 6

(a) A few years ago this question would have been considered one of those intended for really able candidates, but recently most candidates have shown a good basic understanding of protocols and this series was no exception. This is a good example of the way that expectations change from one exam series to another.

(b) The distinction between the logical and physical parts of a protocol should be of equivalent difficulty but it proved far more difficult here. Perhaps this is an area of the syllabus where it would be productive to concentrate on for future series.

(c) Most candidates managed to mention 2 or 3 steps that could be taken but expanding them proved rather more difficult to do. Some of the more powerful systems were rarely seen. It is probably because they will tend to be outside the candidates' experience, but that does not mean that candidates do not need to know about proxy servers and intrusion detection systems.

Question 7

Many candidates are fixed on PC based interfaces that they use themselves, for obvious reasons. However, there are a number of named interfaces in the specification which candidates should be able to describe. The main problems here were a confusion with menu based and form based interfaces. This is no doubt because, once again, the menu based interface featured on the last past paper attempted so that it was at the forefront of the mind when they try to answer here. The natural language interface is not understood by the vast majority of candidates. Most candidates latch on to the word 'language' and describe a voice recognition system. This may be part of a natural language interface but cannot be thought of as synonymous.

Question 8

It was encouraging to see that very few candidates confused OMR and OCR, a distinct improvement on some previous sessions. Most candidates were able to answer the 'why' part of the question, but the description of how it is done was generally sketchy at best.

Question 9

(a) The keyword was 'describe' so there was an extra mark for describing the device. Most candidates scored marks for two devices and for how they would be used but failed to describe them.

(b) A nice simple question to finish on. Questions like this have been asked many times on previous papers and candidates are obviously benefitting from this, however, once again there was evidence of responses which were applicable to previous questions but were not appropriate here.

F452 Programming Techniques and Logical Methods

General Comments

It is in the nature of this unit that, in order to test the candidates' ability to apply programming concepts and principles, questions are asked within the context of a computer program. While some questions may require general answers, candidates should read questions carefully to ensure they recognise when an answer within the context is required. Contrast, for example question 3, "Explain what is meant by an array" to question 4, "Using this example, explain what is meant by a recursive function".

In previous sessions, we have commented on some candidates' inability to define and describe programming concepts which they are clearly familiar with, and hence a necessity to teach some basic theory and definitions, alongside the practical work the candidates do when preparing for this examination. There was some improvement seen in this respect with most candidates gaining the basic marks for definitions and only the more able candidate able to apply it to a particular example or give further details.

The need to set the questions within context poses an interesting problem for examiners. It is inevitable that candidates will vary in their prior knowledge of the context of the problem and this must not be allowed to advantage or disadvantage some candidates. To mitigate this problem examiners use examples which would be in the experience of most candidates, and/or provide as much information as necessary. Candidates should be aware of this, and read questions carefully to ensure they have obtained all the data they need.

Comment on individual questions

Question 1

Part (a) required candidates to apply principles of design to a particular problem by describing the advantages which a drop-down list would have over typing the names of the team. All candidates made a reasonable attempt, and candidates who focused on the two obvious advantages (facilitating data entry and opportunity for validation) did well. In a four mark question it was pleasing to see a majority of candidates attempting to expand their answers and make four distinct points. However, many candidates gave answers that were too vaguely expressed to be given credit such as "there is a limited number of options" which does not quite say the same thing as "only teams which are in the competition can be entered". When answering questions in the context of a given scenario, such as in this case, candidates should endeavour to apply their knowledge to the specific information given. For example, a large number of candidates answered that a drop down list saved space on the interface (presumably, as opposed to listing all the teams). This was not accepted as an advantage because the most natural alternative – a simple text box – would use just about the same amount of space.

Part (b) was well answered with many of the candidates gaining all 4 marks. Most candidates who did not get full marks, did not allow for the fact that the two teams may have drawn (i.e. they gave code of the form IF Team A > Team B THEN Team A wins ELSE Team B wins). They may have done this because the code for detecting a draw is given separately in the question, but failed to appreciate that unless they specified how their code related to the given code, following their logic the program would produce the wrong results (by, in the example above, awarding 4 points to team B and 1 to team A.) Some of the weaker candidates confused the > and < signs. This knowledge is clearly within the specification and was considered a significant part of a correct solution so that candidates were unable to gain full marks without correct use of these signs (eg by using words). A number of candidates explicitly added 0 points to the losing team – while this wasn't wrong and did not disadvantage the candidate, it reflected a possible lack of understanding.

Part (c) (i) was generally well answered, although it is important to remind candidates of the importance, when answering a two mark definition question such as this, to answer in sufficient detail to gain both marks. Weaker candidates confused black-box testing with alpha testing and although they may have made statements which were true (e.g. "black box testing is carried out within the company producing the program") these answers were too vague to be given credit. Most candidates answered part (c) (ii) well. The general comment about candidates' familiarity with the context applies especially to this question. It was felt that every candidate would know enough about the problem to know that the number of goals scored by a team in a football match should be a positive integer. Given that the examiners wanted the candidate to perform a certain amount of analysis on the problem to decide the validation needed, it was considered fair not to explicitly state this fact in the question. Any additional context-knowledge (such as match forfeits, abandoned matches, etc...) was not needed as there was enough material to gain full marks without these. Candidates generally failed because they were too vague, especially about the reason for a test. Many had reasons such as "invalid data". The reason for such a test should clearly identify why the data is invalid, to make it distinct from other invalid data. Similarly, candidates could only gain marks for testing "0-0 draws" as well as testing a draw where goals are scored if they explicitly stated the value of testing 0-0 as the boundary between acceptable and invalid data. Centres should stress to candidates that the examiners are not trying to trick them, and they should seek out the obvious answers in cases such as these. There was ample scope for gaining full marks in this question without resorting to specialist knowledge or test cases which are more difficult to illustrate. Candidates who attempted to bring their knowledge of football competitions into the question often failed to be specific enough to gain full marks, and would have been better advised to stick to the obvious answers. For example, when testing the correct result when the match is forfeited, entering 0-0 was not correct input data – they needed to state specifically that the result is entered as eg Team A: Forfeited.

Part(d) was aimed at E-grade candidates and was well answered. There were, however, too many candidates who did not include a title to their report. On this occasion, the mark scheme still allowed candidates to get full marks with this omission, although centres should be aware that in future sessions, some marks may be ring-fenced for absolutely essential elements of a design such as the date and title of a report. It was pleasing to see that the extraneous annotation seen in previous sessions has diminished considerably. Given that this is in examination conditions, it is not necessary to annotate with details of fonts, colours and UI objects (such as text boxes) nor is it necessary to draw a perfectly neat diagram with ruler and colouring pencils. The purpose is to determine whether, given appropriate software tools, the candidate would design a solution that is fit for purpose. Some annotation, however, is necessary in order to explain elements of the design which cannot be easily drawn. This applied especially to indicating the order in which the teams are listed, as this is specifically required by the question. The question states clearly that no other data about the teams is to be included in the report. A small number of candidates wasted time adding considerably more competition data which was out of the scope of the question.

Question 2

Answers to part (a) were disappointing, with many candidates confusing stepwise refinement with prototyping and iterative development, or with stepping through code and hence gaining 0 marks. For the half who knew the term "stepwise refinement", the third mark in (a) (i) was only obtained by the most able candidates as it required a sufficiently thorough understanding to give a detailed answer without going into the advantages of a modular design which is asked for in (a) (ii). (However, candidates who gave advantages in a (i) were able to have these counted against their mark for (a) (ii)). Note that a mark was not awarded in (a)(i) for stating that when the modules are all programmed they are assembled to create the whole program as this does not explain what stepwise refinement is, but says what happens after you use stepwise refinement. If the question had been “Explain how a top down design can be used to create this programme”, that would have been a perfectly good answer. Part (a) (ii) was answered reasonably well, in particular by the candidates who had answered (a) (i) correctly.

A majority of candidates obtained full marks in 2(b). Some candidates made slight errors or added extraneous lines between the modules. For the candidates who gained 0 or very few marks, it appeared that they were not familiar with the layout of a top down design (most often confusing it with a flow chart). This was especially unfortunate as this question was aimed at lower ability candidates and the structure was clearly described in the question so that the candidate did not have to perform any further refinement. Some candidates did attempt further refinement; this did not disadvantage them as long as the modules described in the question were in their answer. Centres should advise candidates that where further refinement is required, this would be clearly stated in the question.

Part 2(c) was another attempt to use a three mark explanation question to differentiate, and this worked well. While most of the candidates could recall a standard definition of a function they had learnt, only the most able candidates were able to provide sufficient detail for the third mark. Many candidates attempted to expand by describing subroutines in general. As the question was specifically about functions, only one mark was available for stating that it was a subroutine or for describing how it is called like a subroutine, the other two marks reserved for points pertaining specifically to functions. It was very pleasing to see that centres are teaching fundamental principles when using terms such as function, subroutine and procedure, even when the implementation of different languages may use these terms slightly differently (although, as usual, students who explained the terms as they are used in recognised languages were given the benefit of doubt).

The vast majority of candidates knew what a Boolean was in part 2(d) but a good number failed to realise in part 2(e) that the expression in line 06 evaluates to a Boolean. These candidates often stated the value of TotalLength or the value of (TotalLength - 80) rather than the value of (TotalLength > 80). Among the candidates doing it correctly, there was some misunderstanding of the > sign evidenced by the fact that they got the answers the wrong way round, or that they got part (ii) wrong, evaluating 80 > 80 as True.

Part (f) was poorly answered, it was the number and nature of incorrect answers scoring 0 or 1 out of 4 that was rather disappointing. More than half of the candidates gave a completely wrong answer, or more typically, just replaced the word FOR with the word WHILE. Among the better responses, the majority looped while the TotalLength was less than 80 and stopped immediately it exceeded 80. While a function written in this way will produce the same result as the original function with the FOR loop, the question asked the candidate to rewrite the FOR loop as a WHILE loop – it was therefore necessary to have a loop with similar start, end and number of iterations as the original loop. Another common error was to omit initialising the iterator (usually i) of the loop. Once again, allowances were made for non-standard implementation of WHILE loops in known languages such as post condition DO-WHILE loops.

Question 3

The general comment about candidates who fail to describe fundamental concepts which they probably otherwise understand applies to part (a) . The majority of candidates scored 0 or 1 mark in part (i) although most candidates were able to state how an array is declared in part (ii).

Part (b) was well answered with more than half of the candidates gaining the full four marks. This question, which involved performing an operation (assigning a random number) to every element of a 2-dimensional array was intended to be a stepping stone towards the more difficult question in part (c) where candidates needed to use a similar strategy to print every element of the 2-D array. A significant minority of candidates omitted the question completely. Centres should continue encouraging the weaker candidates to attempt these algorithm questions rather than leave them out completely. These long algorithm questions do intend to differentiate the most A/B candidates, but will also have some mark points that are accessible to candidates at the E/U border. Even among the most able candidates, the majority of candidates printed all the data onto one line, without returning to the next line at the end of each row, resulting in a maximum mark of 7.

Part (d) was the quality of written communication question. The best responses, scoring 6 to 8 marks, clearly identified the use of meaningful names as a most important consideration, and showed breadth of knowledge and included several additional strategies. They read the question carefully and were careful to explain both the WHAT needed to be done as well as WHY it was needed and often provided examples. The use of past papers is excellent preparation for any examination, but should be used with care on this unit. Several candidates answered the question from last January (best practice when writing code) and did not get any marks unless they said specifically that variables/constants should be given meaningful names. Centres may have noticed that the amount of lines provided on the page was increased from previous sessions. This was to enable candidates (especially if they had crossed out some of their answer) to be able to continue on the same page without having to answer on additional sheets. The increase in the number of lines was not intended to indicate an increase in the expected response but simply to ensure that candidates would not need to continue on supplementary sheets. Candidates who insisted on filling the page sometimes added vague, irrelevant and even incorrect statements, or repeated the same point several times, reducing the overall quality of communication in their response and consequently, their mark.

Question 4

Many candidates described the ASCII code, rather than the ASCII function in part (a)(i). The question does state twice that it refers to the function. Centres should emphasise the need for candidates to read questions carefully. There was no clear evidence that any candidates were disadvantaged because they had studied a language where the function would be called something completely different (for example “ord” in Pascal). However, this is an opportunity to remind centres that the names of string manipulation functions which will be used in pseudocode in examination questions are clearly listed in the specification. Centres are free to instruct candidates in which ever language they feel will best teach the programming principles in the specification. However, they must refer to such lists of pseudocode terms in the specification and ensure that candidates can relate them to the keywords in the language they have studied. Part (a) (ii) was generally well answered, although some candidates gave answers which are too vague such as “to get the correct answer”. Candidates are not required to memorise ASCII code values, but are required to appreciate the practical use of manipulating character codes. A bit of deduction was needed here, given the purpose of the program, to understand the reason to subtract 65.

Part (b) was generally well answered, with a majority of the candidates gaining 2 or 3 marks out of 3. The candidates were familiar with the definition of recursion although, in some cases did not apply this definition to the function in the question despite being specifically asked to do so. As expected, the vast majority of candidates were able to explain what happens in the function in the terminating case gaining full marks in part (c).

Part (d), on the other hand, which truly tests the candidates’ understanding of recursion and was aimed mainly at higher ability candidates, provided a range of responses which was only to be expected. Many candidates found it difficult to select a format in which to display the result. This was perhaps somewhat influenced by the unusual layout of answer lines in the answer paper.

While there is no requirement to draw a diagram, we would contend that the most ideal way to answer a question such as this is to draw a diagram such as demonstrated in the recursion section of the OCR recommended textbook (or as required by the 2507 tasks in the legacy specification – see, for example, the June 2008 tasks). These diagrams show each function call separately in a box, with arrows connecting them to show where a function has been called recursively; what the return value is; and where the program resumes after the value is returned. Candidates who attempted to use an ordinary trace table produced vague answers which were not able to clearly indicate whether a function had been called again, or whether there had been a branch within the same function call to line 52.

In part (e) most candidates obtained one mark for the return value, but only the most able candidates correctly identified that you need to keep subtracting 9 while Temp > 9. Given that a large number of candidates correctly answered more difficult algorithm and comparison questions, this suggests a general lack of attention to detail and that candidates did not carefully check their answers.

Part (f) was aimed at A-grade candidates. While many candidates did not understand recursion enough to give an answer worthy of any credit here, the better candidates were able to demonstrate real depth of understanding by making appropriate comments about the relative merits and drawbacks of iteration as opposed to recursion. These candidates were rewarded for demonstrating this insight and answers where a valid opinion was expressed and defended in a way which shows deep understanding were given credit.

F453 Advanced Computing Theory

General comments

Most candidates were able to make a good attempt at the paper, though a few were clearly unprepared for content of the examination which faced them. A general problem seemed to be the use of technical terms, such as confusion between disk storage and memory, and use of “file” or “data” to describe almost anything.

Many candidates could have improved their marks if they had been able to use some basic definitions. The Specification covers a range of topics, and it was disappointing to see that some candidates had wasted their time by attempting to learn about additional topics while not knowing those which are clearly listed.

Despite criticisms made in the January 2010 Report, presentation of candidates’ answers remained poor. This issue needs to be addressed: candidates cannot be awarded marks if the examiner cannot read what has been written. Candidates should also know that, once scanned, any work they write in pencil appears the same as that in ink. This is a particular problem when marking any calculations as on some scripts there appeared to be a number of conflicting answers. Candidates should be discouraged from writing a vertical line between each pair of binary digits as these were often indistinguishable from the 1’s and made answers appear to have up to fifteen bits instead of eight.

As before, candidates are reminded that reading and understanding the question is essential. Ignoring instructions is bound to result in the loss of marks.

Comment on individual questions

Question 1

In (a), a number of candidates either did not read the question properly or chose to ignore the fact that it asked about memory management, as they wrote about operating systems in general. Many of those who restricted themselves to a response about memory management thought that it speeded up processing. Some wrote about disk defragmentation. Disk threshing was frequently described as “wearing out” the disk.

Question 2

Many gained good marks for this question, with a number showing at least some understanding of the use of intermediate code. Part (c) was not understood, with the majority of candidates thinking that optimisation removes comments and white space rather than redundant code.

Question 3

This question showed clearly which candidates had prepared properly for the examination. Many gained good marks here.

Question 4

Many candidates gained good marks, with only a few omitting to show working. Some were unable to answer part (c) adequately.

Question 5

Part (a) was answered well. In part (b), it was surprising to see that some were unable to merge the example files, which was only required to make candidates think about their algorithm. Few had any idea about how to write the algorithm. Common mistakes included: not opening files, using the same counter for both files being merged, confusion between < and > signs. It was disappointing to see that many candidates wrote that their assumption was that the files were sorted: this was stated in the question, so they should have known that it would not gain marks.

Question 6

Candidates must read the question. In (a), they were instructed to "tick the boxes where…". The majority incorrectly assumed that only one box on each row of the table was correct, so it was rare to find a candidate with full marks for this part of the question. Part (b) was answered reasonably well, though poor English skills meant that some were unable to explain backtracking adequately.

Question 7

Many gave adequate answers and some good descriptions of local and global variables were given. A common misconception was that parameters are limits or boundaries placed on variables: there appeared to be confusion with validation here. A few candidates ignored instructions about accuracy of communication skills and merely wrote scrappy lists of a few points.

Question 8

Many attempted to give the definition required, but had not learnt correct techniques for doing so. Candidates should be advised to define additional terms, where relevant, if recursive definitions are needed. Those who adopted this method usually gained full marks.

Question 9

A few strong candidates gave good answers here. The majority did not seem to have learnt this topic and there was considerable confusion, especially about indexed addressing.

Question 10

This produced a whole range of different types of answer, with some candidates producing excellent responses. However, some had not learnt this topic. It was disappointing to see that, having identified a many-many relationship in (c) (iv), some candidates could not resolve it. The common errors were to remove the "many" symbol from one end (presumably chosen at random) or to insert a link entity but leave the relationships the wrong way round.

Question 11

This is a new topic for A2, so the question was deliberately fairly simple. Some candidates answered it easily. Others had not learnt this at all and could not even make a sensible guess at the correct names for the diagrams. For many, the most difficult part was giving an example of an object.

F454 Computing Project

The first full session for this unit produced a lot of well crafted projects with well-annotated modular code. It is worth pointing out that a majority of centres evidently understood the requirements and their candidates seemed to be interested and tried hard to produce something worthy of their talents.

Overall, the work from the candidates was well presented, with a contents page and in folders. Where teachers provided supplementary marking notes, this helped significantly, and it was especially helpful when the location of the evidence to support assessments was identified. It should be stressed that the individual candidate cover sheets which give a clear and concise list of the marks for each of the sections of the project must be included for each project and that the inclusion of centre-produced sheets are not an alternative, however valuable they may be.
For some candidates evidence was very hard to locate and it is expected that for such complex reports as these there must be an aid to navigation through them to allow the understanding of the reader. The simplest way to do this is to include a contents page with correct references to page numbers within the report. Centres should consider this to be a necessity of a well presented report.

The poorest projects often failed because of a poor analysis section. Lack of a genuine end user often produced simplistic and superficial analysis and investigation providing little evidence on which to base a design for a system. The interviews often needed to be improved with questions being asked which are related to the organisation and which lead to important facts being discovered which can then be used in the design section. There should be much more succinct and probing interaction between the student/analyst and the client. Students often missed the opportunity to provide further research into the problem by looking at similar systems. Some of the suggestions for alternative solutions were not credible and students should be looking at existing commercial products and seeing exactly what they can do. In some cases there was little evidence that any conversation had ever taken place between the students and an end user.

For the top range of marks the design ought to be good enough to allow a different developer to proceed, and while this was evident in a lot of the work, in some cases the designs were less than useful. The designs need to take the requirements specifications and turn these into measureable objectives that form the basis for a working design. Requirements like ‘user friendly’ do not provide any opportunity to do this and candidates need to provide much more detailed requirements based upon their investigation. The design section should include a complete set of algorithms that can be shown to provide a solution to the stated problem and should also discuss, in some detail, the testing strategy to be followed. Test plans should be designed showing when things will happen such as modular testing, integration testing, beta testing and acceptance testing. A final test sequence that is simply reams of "does this button work" etc is not sufficient evidence for thorough testing of a system.

The development section needs to show the system under development and being tested at each stage. Prototyping is an excellent way of providing this evidence. Endless screen shots without comment are not the best way to convey the processes that have been followed during development.

There were some excellent projects including web-centred solutions combining things like PHP, SQL, Python, Perl, Java, JavaScript as well as well-crafted data management solutions and games. While using ACCESS as a back end database for a VB project is totally acceptable, ACCESS with a bit of VBA does not generally produce top grade projects. The least successful projects used ACCESS with pasted in or slightly edited button wizard code. This is essentially a programming project and there really must be enough data handling code to demonstrate that the candidate can program.

The documentation marks are awarded for good on-screen help and this needs to be identified within the report at some stage. The supplementary user guide should include installation, typical use, troubleshooting, system requirements etc. We do not require a technical guide; this is covered in the report in other places and in the supplementary user documentation.

Some little things make a difference to the moderator’s work load and page numbering, indexes, a breakdown of the marking and location of evidence all help. For example using a footer with the candidate details makes a difference. Once again, while the vast majority of centres complied with all the administrative procedures, missing paperwork and clerical errors do take time to rectify and delay the process.

While these comments are provided to help improve the submissions from centres in future sessions I should like to congratulate the vast majority of centres who provided the moderation team with some excellent, well-organised and carefully planned and prepared material.

OCR (Oxford Cambridge and RSA Examinations)
1 Hills Road
Cambridge
CB1 2EU

OCR Customer Contact Centre

14 – 19 Qualifications (General)
Telephone: 01223 553998
Facsimile: 01223 552627
Email: general.qualifications@ocr.org.uk

www.ocr.org.uk

For staff training purposes and as part of our quality assurance programme your call may be recorded or monitored

Oxford Cambridge and RSA Examinations
is a Company Limited by Guarantee
Registered in England
Registered Office; 1 Hills Road, Cambridge, CB1 2EU
Registered Company Number: 3484466
OCR is an exempt Charity

OCR (Oxford Cambridge and RSA Examinations)
Head office
Telephone: 01223 552552
Facsimile: 01223 552553