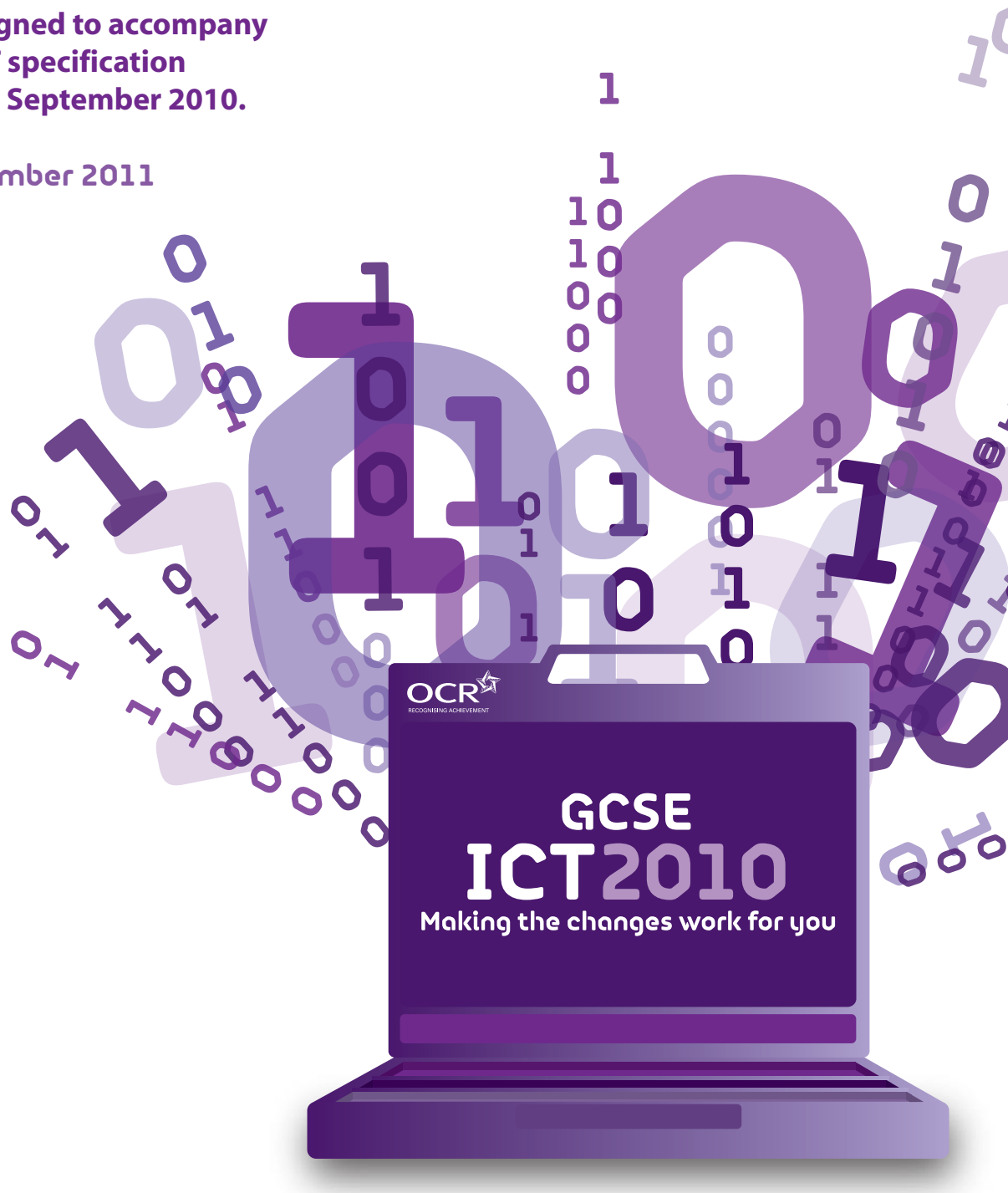
OCR
RECOGNISING ACHIEVEMENT

OCR GCSE in ICT

teachers' handbook – success in B065

This guide is designed to accompany the OCR GCSE ICT specification for teaching from September 2010.

Version 1 September 2011



Contents

1. Unit and Controlled Assessment Overview

1.1 Unit Overview

Unit B065 requires candidates to produce a coded solution to one of the problems set within the controlled assessment document (available on the OCR website). Before commencing the controlled assessment, centres need to spend time teaching and developing candidates' skills so they can produce a solution independently.

Many of the skills required for unit B065 will not naturally form part of the teaching for GCSE ICT and some special provision will have to be made to include these additional skills. The assessment for this unit does, however, include some extra credit for acquisition of these additional skills.

1.2 Controlled Assessment

The concept of controlled assessment is very similar to traditional coursework as candidates produce a solution to a problem along with a write up which follows the system lifecycle, but work is completed under supervised conditions to prevent any malpractice which may occur if work is completed outside of class.

Controlled assessment for unit B065 requires candidates to complete a task set by the examination board within a recommended time frame of 20 hours. Some of this time needs to be allocated to planning and research time, some of which may be done in groups so candidates can share ideas.

A suggested break down of the time allocation is given opposite.

Hours	Activity	Notes
1	Introduction to the controlled assessment tasks	During this stage the teacher will introduce the concept of controlled assessment, the tasks available, time allocation deadlines and working methods to candidates. Candidates will be able to ask questions during this stage.
8	Preparation/research/ collection of evidence	During this stage candidates will collect ideas and materials which will allow them to design and produce an appropriate solution to their chosen task. During this stage group work may take place so candidates can share ideas although the final product must be their **own** work. During this stage teachers can offer guidance to candidates on key things which should be in the folders, interpret marking criteria and answer any questions. A low level of supervision may take place during this stage and candidates will be allowed to do preparatory work without direct supervision however the teacher must be satisfied at all times that it's the candidates own work. This time can be usefully spent identifying the programming skills and concepts required for the solution of the set problem.
12	Producing final piece of work	During this stage work is to be carried out in controlled conditions, which mean under direct teacher supervision. No work can be carried out at home, for example. Candidates will be allowed access to the materials produced during the research stage but centres need to carefully ensure they are preparatory notes and not draft copies of final versions of the product.

There are various problems available and centres should choose from these to match the facilities available and the skills that have been developed. Solutions to the problems will require a coded solution in a suitable programming language and candidates will need to develop and show their ability to use various key features within their solution.

In previous GCSE specifications, templates have been widely available to assist candidates in the structuring and production of evidence - under controlled assessment the **use templates is strictly prohibited** and will be deemed as a case of malpractice if there is any suspicion of use.

B065 is assessed by teachers within a centre using the evidence grids provided, and then a sample of work will be externally moderated. Submission for moderation can be made electronically using the OCR Repository or via postal moderation (paper based or on CD). As products are being made during this it is recommended that centres submit work electronically so products can be seen working and reduces the number of screen shots necessary.

All work which is submitted should be clearly labelled with the centre and candidate's details, unit number (B065) and controlled assessment task for which the work has been produced.

If centres submit work electronically (CD or OCR Repository), each candidate's written work should be compiled into one folder. It may be necessary to compile the program into an executable files before submission to enable the moderator to view the final product.

2. Assessment Topics

B065 requires the candidate to demonstrate various programming skills and these should be taught in advance of taking the controlled assessment

Specification Area	Topic	Rationale	Suggested Activities
Identify and use the three basic programming constructs used to control the flow of a program: sequence, select, iterate	• understand and use suitable select statements • understand and use suitable loops including count and condition controlled loops • use different data types, including Boolean, String, Integer and Real, appropriately in solutions • to problems • define and use arrays as appropriate when solving problems	• Candidates need to have knowledge of the basic programming constructs and should be able to use these effectively and appropriately within their solution to the chosen problem. • It may be that not all techniques are required but this should be justified within the development of the solution and knowledge of these techniques will be essential to make the most appropriate choices.	• Candidates should work with a suitable programming language on simple tasks related to each of these programming features. • Use of SCRATCH may be a good starting point for introducing these but the tasks are designed to be available for a range of programming styles from console programming to visual programming and the choice of task and language should be carefully thought through at an early stage and suitable training put in place for the candidates.
Planning the development of a coded solution to a problem	• identify the information required to solve a problem • produce a plan for the development of the solution • specify the required hardware and software • define the success criteria for later reference during evaluation	• Good program design requires careful analysis of the problem and identification of the techniques and structures required to solve the problem.	• Candidates should take the time to research the problem and identify what is required talking through ideas and potential approaches before going on to design a solution. This is an opportunity to refine the necessary coding skills required for the chosen problem and approach. • Candidates should use this opportunity to research similar problems and identify potential solutions and approaches.

Specification Area	Topic	Rationale	Suggested Activities
Design a coded solution to a problem by developing suitable algorithms and test procedures	• describe how the proposed solution will be fit for purpose • design individual components of the solution • design input and output formats • design an overall solution using suitable algorithms • design testing routines	• The key to a successful coded solution to a problem is careful planning and a clear description of how the solution will work. Clear algorithms are an essential tool for the programmer and the candidate should always know what data is going to be used to test a solution during development.	• Practice writing simple algorithms to define solutions, dry run these algorithms and identify suitable test data that will check the code works. The point of testing is not to show just that a system works but to try and break the system and to find potential faults.
Create a coded solution	• create a coded solution • create systems for input to and output from the solution • create navigational paths and methods • create a working solution	• Candidates need to create the solution showing how each part of the solution contributes to the whole solution. Annotation is a key to clear readable and maintainable code and candidates should clearly annotate the code to explain how it works.	• Working at first with simple tasks candidates should develop the skill of writing then modifying code to do more complex tasks, testing at each stage.
Test the solution	• test the solution they have produced • have potential users test their solution and test solutions developed by others	• Candidates must test the solution to show functionality and how it matches the design criteria. They should have identified in their analysis what the solution should do and, through the test plan, how they would know that the solution did what was required • They should identify success and any limitations, describing ways the solution can be improved.	• Taking simple tasks and testing as they develop will get them into the habit of prototyping. • Candidates should get into the habit of saving versions of their code not only to be able to retrace their steps back to a working version, but also to demonstrate the development and testing process for the moderator.

Specification Area	Topic	Rationale	Suggested Activities
Evaluation	• use the results of testing and identify the limitations of their solution • use the results of testing and recommend possible improvements to their solution • evaluate the solution with regard to purpose • evaluate the solution with regard to the success criteria • improve their solution	• There is little point in testing if it does not lead to an improvement in the solution. We expect the solution to have flaws, we expect the best candidates to find these flaws and the very best to be able to fix or recommend a way that they might be fixed. Testing is only effective when it finds the errors, ineffective testing will simply test basic functionality in ideal conditions and fail to find the situations under which a solution does not work as expected.	• Candidates need to be aware that testing is there to find errors and not simply to prove that a system works under ideal conditions, get them to try and break any simple solution they develop. They will be credited for finding flaws if they exist and for suggesting how these might be fixed, we do not expect flawless solution.

3. Assessing B065

Work for B065 should be marked on mark sheets produced by OCR, which are available from the OCR website. Each section of the work is spilt into three different mark bands with criteria becoming more challenging as you move upwards through the bands. Marks for this unit should be awarded using the best fit principle.

3.1 Step 1- Which band?

Assess quality of the work produced using the descriptors on the mark sheet and decide which of the three mark bands best describes the work. Marking should be positive, rewarding achievement rather than penalising failure or omissions. The award of marks **must be** directly related to the marking criteria. Teachers should use their professional judgement in selecting the band descriptor that best describes the work.

3.2 Step 2 – Which mark within the band?

Once a mark band has been determined, a mark within the band needs to be awarded. The most appropriate mark within the band should be chosen using this guidance:

- where the work convincingly meets the statement, the highest mark should be awarded
- where the work adequately meets the statement, the most appropriate mark in the middle range should be awarded
- where the candidate's work just meets the statement, the lowest mark should be awarded.

Teachers should use the full range of marks available to them and award full marks in any band for work that fully meets that descriptor - this is work that is 'the best one could expect from candidates working at that level'. Where there are only two marks within a band the choice will be between work which meets the statement in most respects and work that just meets the statement. For wider mark bands the marks on either side of the middle mark(s) for 'adequately met' should be used where the standard is lower or higher than 'adequate' but not the highest or lowest mark in the band.

If centres have large entries and/or multiple internal assessors it is vital that internal standardisation takes place to ensure all candidates are assessed to the same standards.

4. Criteria Amplification

<table>
<tr><td colspan="2">Analysis Section (0-10 Marks)
This part of the work requires candidates to find and review existing solutions to problems similar to the one they are going to complete for their controlled assessment. The solutions hopefully will give candidates ideas and inspiration to design their own product. Once several solutions have been investigation candidates then need to write a design specification for their product where they set out what will be in there product. Candidates should also think about the programming constructs they are likely to need and identify examples of how these can be utilised in their own solution. Evidence for this section is likely to take the form of a word processed document and some code snippets.</td></tr>
<tr><td rowspan="3">Candidates will need to complete some analysis of the task and identify the requirements.

Candidates will need to look at existing solutions and identify key features that will be of use to them in their own designs.

Candidates will need to identify the programming constructs they are likely to need and identify suitable examples of how these can be used effectively.

Candidates need to consider any specialist hardware or software requirements for their solution (there may be none).</td><td>0-3 marks
Typically candidates will have carried out some limited research into similar problems but much of this will not be directly relevant. Planning will be limited identifying only the most obvious requirements for the end user identified within the task.</td></tr>
<tr><td>4-7 marks
The research will be more extensive and related to the set task. The information identified will have some relevance to the set task and will provide some ideas that can be followed up. At this level candidates will have some idea of how they will decide if their solution was successful in general terms. The research will have been analysed to identify relevant features and the candidates are likely to have identified what they need to find out in order to complete the task. There will be an outline plan for completing the task.</td></tr>
<tr><td>8-10 marks
At this level the research will be focussed on the task requirements and a range of relevant sources will have been identified and analysed carefully to inform their work. The success criteria will be measureable rather than general and it will be clear how these candidates will identify if they have successfully completed the task. There will be a detailed plan showing how the solution will be completed with timings/stages.</td></tr>
</table>

<table>
<tr><td colspan="2">Design Section (0-12 Marks)
Within this section candidates explain exactly what they intend to make and design the screen and other elements which make the product. For higher marks somebody else will be able to make the product from scratch using the designs produced. Testing strategies will also be covered within this section with more able candidates writing a full test plan for use later in the project. Evidence for this task is likely to be a word processed document.</td></tr>
<tr><td rowspan="3">This section will contain descriptions of how the solution solves the problem together with any relevant screen designs, data/file structures.

The processes will be defined using algorithms in some form, for example flow charts, pseudo-code, block diagrams etc.

There will be a description of the planned testing to be completed during and after development.</td><td>0-4 marks
Typically there will be some brief comments on the solution and the proposed method of solution. The design will be limited to some layouts and to outline algorithms possibly as flowcharts, but these are unlikely to explain the data flow and detail of the solution. Planning will be limited to a basic functional test plan for a limited range of the variables.</td></tr>
<tr><td>5-8 marks
There will be more detailed algorithms explaining data flows and some of the detail in the solution, but there may be omissions or minor errors in these. Designs may often be replaced by code produced for the solution and not pseudo code developed as part of the design process. There will be some thought given to the layout and to user interaction with the system together with a test plan that covers many of the success criteria.</td></tr>
<tr><td>9-12 marks
Work at this level will include detailed algorithms, pseudo code and some evidence that the algorithms have been checked to show that they form a working solution to the problem. There will be careful consideration of user interaction with the system and the user interface. There will be detailed testing plans with data and expected outcomes that can be used during development and data that can be used post development to measure against the success criteria.</td></tr>
</table>

<table>
<tr><td colspan="2">Use of coding features (0-11 Marks)
This section is where candidates are credited for the extra skills they have had to develop in order to complete this unit. The marks are awarded for their use of coding features and should be established from within their solutions. We do not require a section illustrating their use of each feature, merely evidence from within the code presented as part of the report.</td></tr>
<tr><td rowspan="3">This section is assessed on the basis of the code produced. In order to do well in this section candidates need to utilise an appropriate range of coding features and to annotate their code to show how it works.</td><td>0-4 marks
Typically there will be evidence of some coding using a limited range of features. Some of the standard structures will have been used but the code may only solve a small part of the problem. Variable names may not be sensibly named and programming techniques are unlikely to be used effectively or efficiently in most cases.</td></tr>
<tr><td>5-8 marks
We care likely to see a reasonable attempt at solving the problem. Many aspects are likely to work as required but there may be omissions or minor errors. The techniques will have been used effectively in many cases and variables will be chosen and named appropriately. The use of structures may well work under typical conditions but inefficiently and be prone to issues under rigorous testing.</td></tr>
<tr><td>9-11 marks
The standard programming techniques will be used effectively and efficiently to produce a solution. The structures will be appropriate and all variables will be appropriately named. For example candidates at this level will use CASE statements rather than multiple IF THENs, they will use arrays rather than sets of variables for similar data.</td></tr>
</table>

Development of the overall solution (0-7 Marks) Once again candidates will gain marks from showing the stages in the development of their solution with evidence of versioning and intermediate testing during the development stage.	
This section is assessed on the basis of evidence of the development process, not on evidence that the solution exists. Candidates must show the stages in the development with intermediate testing and should keep versions of their program as it is developed.	0-3 marks There is unlikely to be very much evidence of intermediate versions and testing just some evidence of the solution with comments or partial solutions related to the design.
	4-5 marks At this level candidates will have some explanation of the process and some evidence of intermediate stages. The solution will have some functionality but may have some problems, inefficiencies or limited functionality.
	6-7 marks There will be detailed evidence of the process including intermediate versions, testing evidence and evidence of further development following testing. All the stages will be explained with a clear commentary and there will be a functional and efficient solution to the problem.

Testing (0-10 Marks) There will be evidence of testing during development and post development. Marks in this section are awarded for the overall value of the testing and not just evidence found under the heading 'testing'. Developing code is a circular process and work does not fall neatly into discrete sections, this should be accounted for when assessing the work.	
This section is assessed on all evidence of testing found at any stage within the process.	0-3 marks There will be evidence of output from the system and possibly some evidence that others have used the system. Testing will be limited to very basic functionality in typical conditions and for that part of the system that is known to work.
	4-7 marks At this level the program is likely to be almost fully functional and there should be evidence to show that the system has been tested with typical and atypical data / input. Expected outcomes should be identified and commented upon. Testing should cover the key elements of the design criteria and there will be evidence that the system has been tested under more than one 'safe' situation, eg on another computer.
	8-10 marks The system should have been tested at various stages in the development and should cover all aspects of the design. The purpose of testing is to try and break the program and at this level it should be clear that candidates have used data and situations that test the program in extreme conditions. There will be testing in a wide variety of situations to show that the system is robust and meets the success criteria. If this rigorous testing does identify faults under extreme conditions the candidate should not be penalised for knowing this, that is the purpose for testing.

<table>
<tr><td colspan="2">Evaluation (0-10 Marks)
This section should bring together the evidence and it is time for the candidate to reflect on the process and the success of the project compared to the original aims. It is not about saying ‘I would have worked harder’ or to place blame on the server for running slowly. It is about looking at the evidence from the testing stages to compare the finished product with the success criteria and the original requirements.</td></tr>
<tr><td rowspan="3">This section is likely to be a report written at the end of the process. Candidates should gather together all the evidence they have saved during the process, test output, diaries, commentaries, versions of the software and. Perhaps, screen-capture evidence of the system in action. They should write a report listing the success criteria ad comparing this to the evidence they have to show how successful, or otherwise, they were. This section will also be used to assess the quality of the candidates’ written communication.</td><td>0-3 marks
Some limited comments on the success or otherwise of the solution but not specifically related to the original criteria or testing evidence.</td></tr>
<tr><td>4-7 marks
A description of what the system can do, some evidence to support this and all related to the original design requirements. There may be some comments on limitations but with little attempt to discuss these or potential solutions / reasons.</td></tr>
<tr><td>8-10 marks
A full evaluation of the success criteria clearly related to test evidence. The limitations of the system will be identified and discussed with potential solutions or potential extensions to the solution based on the results of testing.</td></tr>
</table>

The evidence required to assess the work will include a collection of written reports for the research, analysis, design and evaluation. There will also be written commentaries on the development and testing processes supported by electronic evidence of thee code at various stages and evidence of test output, possibly as screen-capture video or screen shots. The evidence must include the final version of the program and this should be checked to be sure it will work away from the school system. Submission can be electronic via the OCR Repository or via a postal entry on CD. The second method allows for any paper based evidence to be included without the need for scanning.

www.ocr.org.uk
OCR customer contact centre

Vocational qualifications
Telephone 024 76 851509
Facsimile 024 76 851633
Email vocational.qualifications@ocr.org.uk

J461/J061/SUP/10