Advanced Higher Computing Science

Draft National Course Assessment Specification

Valid from August 2015

This edition: December 2012, draft version 1.0

Please refer to the note of changes at the end of this Course Assessment Specification for details of changes from previous version (where applicable).

Contents

Course outline

Course title:	Advanced Higher Computing Science
SCQF level:	7 (32 SCQF credit points)
Course code:	to be advised
Course assessment code:	to be advised

The purpose of the Course Assessment Specification is to ensure consistent and transparent assessment year on year. It describes the structure of the Course assessment and the mandatory skills, knowledge and understanding that will be assessed.

Course assessment structure

Component 1 — project	90 marks
Component 2 — question paper	60 marks
Total marks	**150 marks**

This Course includes eight SCQF credit points to allow additional time for preparation for Course assessment. The Course assessment covers the added value of the Course.

Equality and inclusion

This Course Assessment Specification has been designed to ensure that there are no unnecessary barriers to assessment. Assessments have been designed to promote equal opportunities while maintaining the integrity of the qualification.

For guidance on assessment arrangements for disabled learners and/or those with additional support needs, please follow the link to the Assessment Arrangements web page: www.sqa.org.uk/sqa/14977.html.

Guidance on inclusive approaches to delivery and assessment of this Course is provided in the Course Support Notes.

Assessment

To gain the award of the Course, the learner must pass all of the Units as well as the Course assessment. Course assessment will provide the basis for grading attainment in the Course award.

Course assessment

SQA will produce and give instructions for the production and conduct of Course assessments based on the information provided in this document.

Added value

The purpose of the Course assessment is to assess added value of the Course as well as confirming attainment in the Course and providing a grade. The added value for the Course will address the key purposes and aims of the Course, as defined in the Course Rationale. It will do this by addressing one or more of breadth, challenge, or application.

In this Course assessment, added value will focus on the following:

- breadth — drawing on knowledge and skills from across the Course
- challenge — requiring greater depth or extension of knowledge and/or skills
- application — requiring application of knowledge and/or skills in practical or theoretical contexts as appropriate

Through the Units, learners will develop computing science skills, programming and application development techniques, and knowledge and understanding of key concepts in the project management lifecycle, and have the opportunity to explore specific areas of interest within computing science disciplines.

The added value consists of the following.

To achieve success in the Course, learners must show that they can **apply** the knowledge and skills developed through the Units, in both practical and theoretical contexts. Added value will be assessed through a project and a question paper.

The project requires learners to demonstrate aspects of challenge and application in a practical context. Learners will **apply** knowledge and skills from the Units to plan, design, implement, evaluate and report on a solution to solve an appropriately challenging practical computing problem.

The question paper requires learners to demonstrate aspects of breadth and application in theoretical contexts. Learners will apply breadth of knowledge from across the Units, and depth of understanding, to answer appropriately challenging questions in computing science contexts.

Grading

Course assessment will provide the basis for grading attainment in the Course award.

The Course assessment is graded A–D. The grade is determined on the basis of the total mark for all Course assessments together.

A learner's overall grade will be determined by their performance across the Course assessment.

Grade description for C
For the award of Grade C, learners will have demonstrated successful performance in all of the Units of the Course. In the Course assessment, learners will typically have demonstrated successful performance in relation to the mandatory skills, knowledge and understanding for the Course.

Grade description for A
For the award of Grade A, learners will have demonstrated successful performance in all of the Units of the Course. In the Course assessment, learners will typically have demonstrated a consistently high level of performance in relation to the mandatory skills, knowledge and understanding for the Course.

Credit
To take account of the extended range of learning and teaching approaches, remediation, consolidation of learning and integration needed for preparation for external assessment, six SCQF credit points are available in Courses at National 5 and Higher, and eight SCQF credit points in Courses at Advanced Higher. These points will be awarded when a Grade D or better is achieved.

Structure and coverage of the Course assessment

The Course assessment will consist of two Components: a project and a question paper.

Component 1 — project
The purpose of the project is to assess practical application of knowledge and skills from across the Course to develop a solution to an appropriately challenging and complex computing-based problem. It will assess learners' skills in planning and designing a solution to a problem, implementing and testing a solution, and evaluating and reporting on that solution.

The assignment will have 90 marks (60% of the total mark).

The project will be based on a meaningful and appropriately challenging task requiring challenge and application.

The project should clearly demonstrate application of knowledge and skills, at an appropriate level, from both the *Software Design and Development* and *Information System Design and Development* Units (as defined in the 'Further mandatory information on Course coverage' section of this document).

Marks will be awarded for:

- producing a formal project plan
- producing a detailed design proposal
- iterative development of solution
- final testing and evaluation
- presenting/communicating the solution

Evidence should include:

- the project plan and design proposal
- the completed solution
- a record of progress through the project, including reflective commentary
- a report on the testing and a qualitative evaluation of the solution
- presentation/communication of the solution

The presentation or communication of the solution can be in any suitable format: visual, oral, manual or electronic.

Component 2 — question paper
The purpose of the question paper is to assess breadth of knowledge from across the Course, depth of understanding, and application of this knowledge and understanding to answer appropriately challenging questions.

This question paper will give learners an opportunity to demonstrate the following skills, knowledge and understanding:

- the ability to apply computational thinking to understand problems across a range of contexts
- analysing complex problems within computing science across a range of contemporary contexts
- the ability to communicate how a well-structured, complex modular program works
- the ability to communicate complex computing science concepts clearly and concisely, using appropriate terminology
- in-depth knowledge and understanding of key aspects of contemporary information system project planning and management
- knowledge and understanding of contemporary programming paradigms

The question paper will have 60 marks (40% of the total mark).

Approximately half of marks will be awarded for questions (or parts of questions) related to *Software Design and Development*, and half for questions (or parts of questions) related to *Information System Design and Development.*

The question paper will consist of structured and extended response questions, and will give learners an opportunity to demonstrate application of knowledge and understanding to answer appropriately challenging context-based questions by drawing on and applying knowledge from the table provided in the 'Further mandatory information on Course coverage' section at the end of this Course Assessment Specification.

Setting, conducting and marking of assessment

Question paper

This question paper will be set and marked by SQA, and conducted in centres under conditions specified for external examinations by SQA. Learners will complete this in 1 hour and 30 minutes.

Controlled assessment — project

The project is:

- set by centres within SQA guidelines
- conducted under some supervision and control

Evidence will be internally marked by centre staff in line with SQA marking instructions.

All marking will be quality assured by SQA.

Setting the assessment

Set by centres within SQA guidelines.

- The specification for the project will be agreed between the learner and the teacher/lecturer.
- The project will be a meaningful and appropriately challenging task, which should clearly demonstrate application of knowledge and skills, at an appropriate level, from both the *Software Design and Development* and *Information System Design and Development* Units (as defined in the 'Further mandatory information on Course coverage' section of this document).

Conducting the assessment

Conducted under some supervision and control.

- The project will be carried out under open book conditions, but supervised to ensure that the work presented is the candidate's own work.
- The teacher/lecturer may give learners limited guidance to help them progress through each stage of the project; however, the learner is expected to work independently.
- The project is designed to discriminate between candidates, and therefore would be expected to provide a wide range of marks. Stronger candidates should be able to complete the project successfully with minimal support and guidance. Weaker candidates may not be able to complete all aspects of the assignment to a satisfactory standard.

Further mandatory information on Course coverage

The following gives details of mandatory skills, knowledge and understanding for the Advanced Higher Computing Science Course. Course assessment will involve sampling the skills, knowledge and understanding. This list of skills, knowledge and understanding also provides the basis for the assessment of Units of the Course.

The Course assessment will require learners to draw on and apply knowledge of any of the concepts listed below. This table should be read in conjunction with the descriptions of the question paper and assignment.

Software Design and Development	
Programming paradigms	object-oriented (object, encapsulation, method, property, class, inheritance, UML) imperative (variables, sequence, selection, iteration, modularity) declarative (facts, rules, inferencing, recursion, knowledge base) concurrent programming
Constructs and computational principles	Exemplification and implementation of the following constructs: ♦ reading and writing data to and from sequential files ♦ reading and writing data to and from existing databases Description and exemplification of the following: ♦ abstraction ♦ decomposition ♦ generalisation ♦ pattern recognition ♦ algorithms
Data types and structures	records, linked lists arrays of records and/or arrays of objects *object-based structures (trees)*
Algorithm specification	binary search selection with two lists bubble sort quicksort data/file handling
Design and development methodologies	Prototyping Data flow/entity relationship/state transition diagrams UML use case and class/structure diagram Other contemporary methodologies

Information System Design and Development	
Human Computer Interaction (HCI)	User-centred design, capturing user requirements User interface design, user interface models (menu-driven, textual, graphical, sensory, predictive, adaptive, natural language interaction, multimodal) Usability and user testing Accessibility: W3C, WAI
Testing and documenting (for both software and information system design and development)	Module, component and beta (acceptance) testing verification and validation user guide documentation system/program design documentation code refactoring quality assurance
Project planning and management	Description and exemplification of iterative project lifecycle: **Research**: feasibility studies, user surveys **Planning:** project, scheduling, budget, resources **Requirements** analysis: ♦ business needs, budget, scope, constraints ♦ functional, system attributes, capabilities, characteristics and/or qualities **Specification**: architecture, software/hardware, technical **Design**: modelling approaches (business, data, process), interface design **Implementation**: construction, integration, deployment **Testing** and **documentation** **Evaluation**: usability, quality, goal-tracking **Maintenance**: upgrades, corrective, adaptive **Project planning tools**: Project proposal Online/collaborative project management tools Gantt charts Feasibility studies User surveys System specification and operational requirements Design diagrams/data flow diagrams Test specification and test data Evaluation tools
Roles of computing	Research (academic, market)

professionals	Analysis (business, systems, usability) Design (graphics, interface, media, user-experience, program, system, web) Development/engineering (program, systems, software, database, web, application, code) Project management/client liaison Testing Technical support Information management Quality management Customer/communications liaison Other professional roles (optimisation, digital marketing)
Legal and ethical implications	Copyright, IPR, trademark, patent Social control Man v machine?
Environmental, economic and societal impact	Positive and negative aspects Energy (use and re-use, data centres, mobility, low carbon equipment, battery technologies) Value creation/value chain Internet: marketing (web, e-mail, txt), tracking, analytics Social media/networking Atomisation (job loss/creation)

Administrative information

Published: December 2012 (draft version 1.0)

Superclass: to be advised

History of changes to Course Assessment Specification

Course details	Version	Description of change	Authorised by	Date

Note: You are advised to check SQA's website (**www.sqa.org.uk**) to ensure you are using the most up-to-date version of the Course Specification.