National 5
Course Assessment
Specification

National 5 Computing Science Course Assessment Specification (C716 75)

Valid from August 2013

Revised: April 2016, version 1.3

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

Course outline

Course title:	National 5 Computing Science
SCQF level:	5 (24 SCQF credit points)
Course code:	C716 75
Course assessment code:	X716 75

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 — question paper	90 marks
Component 2 — assignment	60 marks
Total marks	**150 marks**

This Course includes six 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: http://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 software and information system design and development skills, and knowledge and understanding of key computing science concepts and processes in a variety of contexts.

This added value consists of the following.

To achieve success in the Course, learners must show that they can **apply** knowledge and skills developed through the Course to solve problems, in both practical and theoretical contexts.

The assignment requires learners to demonstrate aspects of challenge and application in a practical context. Learners will **apply** knowledge and skills from the Course to solve an appropriately challenging, practical computing science 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 Course 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 question paper and an assignment. The question paper will have two Sections.

Component 1 — 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:

- applying aspects of computational thinking across a range of contexts
- analysing problems within computing science across a range of contemporary contexts
- the ability to communicate how a program works
- communicating understanding of key concepts related to software design and development, and information system design and development, clearly and concisely using appropriate terminology
- understanding of the legal implications and environmental impact of contemporary information system technologies
- applying computing science concepts and techniques to create solutions across a range of contexts

The question paper will have 90 marks out of a total of 150 marks. This is 60% of the overall marks for the Course assessment.

Approximately 50% of the marks will be awarded for questions related to Software Design and Development. These will include questions sampling from the following areas:

- computational constructs and concepts:
  - explaining code
  - writing code
  - data types and structures
- software development — design, testing, documentation
- low-level operations and computer architecture

Approximately 50% of the marks will be awarded for questions related to Information System Design and Development. These will include questions sampling from the following areas:

- database design, structures, links and operations
- website design, structures and links
- coding
- media types, including file size calculations
- information system development — purpose, features, user interface, testing
- technical implementation (hardware, software, storage, networking/connectivity)
- security, legal and environmental issues

However, many concepts are relevant to both software and information system design and development, so some questions will relate to both of these broad areas.

Questions assessing understanding and application of programming skills will be expressed using SQA standardised reference language, which may include the following terms:

Variable types:	INTEGER, REAL, BOOLEAN, CHARACTER
Structured types:	ARRAY, STRING
System entities:	DISPLAY, KEYBOARD
Assignment:	SET … TO …
Conditions:	IF .. THEN .. (ELSE) ... END IF
Conditional repetition:	WHILE … DO … END WHILE
	REPEAT … UNTIL …
Fixed repetition:	REPEAT … TIMES … END REPEAT
Iteration:	FOR .. FROM .. TO .. DO .. END FOR
	FOR EACH … FROM … DO … END FOR EACH
Input/output:	RECEIVE … FROM …
	SEND … TO ..
Operations:	-, +, *, /, ^, mod, &
Comparisons:	=, ≠, <, <=, >, >=
Logical operators:	AND, OR, NOT
Pre-defined functions:	id(parameters)

Where learners are required to answer by writing code, answers may be expressed using any form of pseudocode, any other design notation or any programming language; marks will be awarded for demonstrating understanding, not for correctness of syntax.

Note: Further information on the SQA standardised reference language can be downloaded from the SQA website.

The question paper has two Sections.

Section 1 will have 20 marks and will consist of short answer questions.

This Section will give learners an opportunity to demonstrate breadth of knowledge from across the topics specified in the tables provided in the 'Further mandatory information on Course coverage' section at the end of this Course Assessment Specification.

Questions in this Section will:

- assess breadth of knowledge
- require some calculations

Section 2 will have 70 marks and will consist of structured questions.

This Section 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.

Questions in this Section will:

- assess application of understanding with very few questions requiring direct recall of knowledge
- have balanced sampling across the Course
- consist of questions set in meaningful contexts, that require learners to provide some descriptions and explanations, and may involve some calculations
- include some structured questions that draw on understanding from two or more topics; some parts of these questions may require integration (combining understanding from two or more topics)

Questions related to programming will be presented using SQA standardised reference language.

A proportion of marks will be available for more challenging questions, which could require integration, detailed descriptions or explanations, writing code, and/or analysis, comparisons and evaluations.

For more information about the structure and coverage of this Component of the Course assessment, please refer to the Question Paper Brief.

Component 2 — assignment
The purpose of the assignment is to assess practical application of knowledge and skills from across the Course to develop a solution to an appropriately challenging computing science problem. It will assess learners' skills in analysing a problem, designing, implementing and testing a solution to the problem, and reporting on that solution.

The assignment will have 60 marks out of a total of 150 marks. This is 40% of the overall marks for the Course assessment.

Time will be required for:

- preparation for the assignment, which could include considering exemplar assignments and practising required skills
- carrying out the stages of the assignment, with assessor guidance and support
- assessing the process and completed solution

The assignment should clearly demonstrate application of knowledge and skills, related to the design and development of software and information systems (as defined in the 'Further mandatory information on Course coverage' section of this document).

Guidelines for the assignment will include a list of questions/tasks/prompts which will lead learners through the assignment in clear stages.

Marks will be awarded for:

- Analysing the problem
- Building a solution (designing, implementing, testing)
- Reporting on the solution

Evidence should include:

- the completed solution
- a record of progress through the assignment (such as an informal electronic log, blog or diary maintained by the learner)
- a short report on the solution (in written, electronic and/or oral form)

Setting, conducting and marking of assessment

Question paper

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

Controlled assessment — assignment

The assignment is:

- set by SQA
- 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 SQA.

A bank of assignments will be provided, and there will be choice from this bank.

Conducting the assessment

Conducted under some supervision and control.

The assignment will be carried out under open book conditions, but supervised to ensure that the work presented is the learner's own work.

The assessor may give learners support and guidance to help them progress through each stage of the assignment; where any significant amount of support is provided, this should be reflected in the marks awarded.

The assignment is designed to discriminate between learners, and therefore would be expected to provide a wide range of marks. Stronger learners should be able to complete the assignment successfully with minimal support and guidance. Weaker learners may not be able to complete all aspects of the assignment within a reasonable time, or may require significant assistance, and so would achieve a lower total mark.

Once the assignment has been completed and assessed, it must not be returned to the learner for further work to improve their mark.

Further mandatory information on Course coverage

The following gives details of mandatory skills, knowledge and understanding for the National 5 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 (question paper and assignment) will require learners to draw on and apply knowledge of any of the topics listed below. This table should be read in conjunction with the descriptions of the question paper and assignment.

Component 1 — 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. The question paper Component of Course assessment will require learners to draw on and apply knowledge and understanding of a sample of all the topics listed in **both tables** below.

Software Design and Development	
Computational constructs	♦ Description, exemplification and implementation of the following constructs: — expressions to assign values to variables — expressions to return values using arithmetic operations (+, -, *, /, ^, mod) — expressions to concatenate strings and arrays using the & operator — use of selection constructs including simple and complex conditional statements using logical operators (AND, OR, NOT) — iteration and repetition using fixed and conditional loops — pre-defined functions (with parameters) including o Random o Integer o Round
Data types and structures	♦ Description, implementation and exemplification of the following data types and structures: — character — String — numeric (integer and real) variables — Boolean variables — 1-D arrays
Testing and documenting solutions	♦ Description, identification, exemplification and implementation of normal, extreme and exceptional test data. ♦ Description and identification of syntax, execution and logic errors. ♦ Description, identification and exemplification of the readability

	of code including: — internal commentary — meaningful identifiers — indentation — white space
Algorithm specification	♦ Description, exemplification and implementation of standard algorithms, including: — input validation
Design notations (also applies in information system design and development)	♦ Description and identification of structure diagrams, flowcharts and pseudocode to solve problems. ♦ Exemplification of pseudocode to solve problems.
Low-level operations and computer architecture	♦ Explanation of the need to translate high-level program code to binary (machine code). ♦ Comparison of interpreters and compilers. ♦ Description and exemplification of the use of binary to represent positive integers. ♦ Conversion from Binary to decimal and vice-versa. ♦ Description of floating point representation of real numbers using the terms mantissa and exponent. ♦ Description of ASCII code (7-bit) used to represent characters. ♦ Description of the vector graphics method of graphic representation. ♦ Description of the bit-mapped method of graphics representation. ♦ Describe the purpose of the Basic computer architecture components and how they are linked together including: — processor (registers, ALU, control unit) — memory — buses (data and address) — interfaces
Contemporary developments	♦ Exemplification of trends in the development of: — software development languages — software development environments — their editing features — high-level code translation and execution

Information System Design and Development	
The following mandatory generic concepts and vocabulary may be applicable to a range of information systems types and contexts (including databases, websites, games, mobile applications, kiosk systems).	
Structures and links (database)	♦ Implementation of a relational database with two linked data tables. ♦ Advantages of relational database over flat-file databases. ♦ Description, exemplification and implementation of primary keys and foreign keys. ♦ Description and exemplification of field types (text, number, date, time, graphic, object, link, Boolean). ♦ Description and exemplification of validation including: — presence check — restricted choice — field length — range ♦ Description and exemplification of database operations search, sort (on multiple fields) and calculations. ♦ Description, exemplification and implementation of good design to avoid data duplication and modification errors (insert, delete, update).
Structures and links (web-based)	♦ Description of website, page, URL in relation to a web-based information system. ♦ Description and implementation of hyperlinks (internal, external), relative and absolute addressing. ♦ Understand the need for, and exemplify, simple navigation within web-based information systems. ♦ Description and features of web browsers and search engines.
User interface (also applies in software design and development)	♦ Description of requirements for a good user interface including: — visual layout — navigation — selection — consistency — interactivity — readability
Media types	♦ Description of Standard file formats and their benefits. ♦ Know a range of standard file formats for different media types including: — The text standard file formats txt, rtf — The audio standard file formats wav, mp3 — The graphic standard file formats jpeg, gif, png, svg — The video standard file formats mp4, avi

	♦ Describe and exemplify the factors affecting file size and quality, including resolution, colour depth, sampling rate. ♦ Calculation of file size for colour bitmap. ♦ Description of the need for compression.
Coding	♦ Description and identification of coding to create and modify information systems including JavaScript mouse events. ♦ Description, exemplification and implementation of coding to create and modify information system including the use of: — HTML with the tags for: o Document o Links o Graphics
Testing	♦ Description and exemplification of testing information systems including: — links and navigation work correctly — matches user interface design — media such as text, graphics and video display correctly
Purpose, features, functionality, users	♦ Description of purpose of an information system. ♦ Description of the features and functions of an information system. ♦ Description of types of users of information systems including: — expert — novice ♦ Description of age-range of users of information systems.
Technical implementation (hardware requirements)	♦ Description and exemplification of the appropriate type of hardware required for a specific information system including: — input and output devices — processor type and speed (Hz) — memory capacity (RAM)
Technical implementation (software requirements)	♦ Describe the purpose of an operating system including: — controlling peripherals — running software — HCI ♦ Understand the features of web browsers including: — OS support — privacy modes — Ad filtering — page zooming

	♦ Description and exemplification of the appropriate type of software required for a specific information system including: — type of application — operating system
Technical implementation (storage)	♦ Comparison of local versus cloud storage ♦ Comparison of built-in versus portable storage Comparison of different interface types and their data transfer speeds including: — Firewire — USB — Bandwidth ♦ Description of different types of storage devices and their media in terms of functionality and capacity (in appropriate units) including: — magnetic — optical — solid state ♦ Description and exemplification of the appropriate type of storage required for a specific information system including: — type of device — capacity — interface type
Technical implementation (networking/connectivity)	♦ Description and comparison of the following transmission media in relation to data speeds and ease of use: — wired — optical — wireless ♦ Description and exemplification of hardware required for network connectivity including: — Network Interface Card — router — hub — switch ♦ Description and exemplification of the appropriate type of network connection required for a specific information system including: — hardware — transmission media
Security risks	♦ Description and identification of the following security risks: — phishing — keylogging (software and hardware) — virus — online fraud — identity theft

Security precautions	♦ Description and exemplification of anti-virus software. ♦ Description and exemplification of good practice in passwords settings. ♦ Description and exemplification of biometrics including: — retina scanning — finger prints — palm prints — face recognition ♦ Description and exemplification of firewalls.
Legal implications	♦ Description, identification and implications for individuals and businesses of the Computer Misuse Act including: — use of software and hardware to access data unlawfully — Impairing of operation of computer systems ♦ Description, identification and implications for individuals and businesses of the Data Protection Act including: — data in electronic transmission — prior consent of data subject — export of data ♦ Description, identification and implications for individuals and businesses of the Copyright, Designs and Patents Act (plagiarism) including: — copyright of computer software — software piracy — web content — text, graphics, video, audio ♦ Description, identification and implications for individuals and businesses of the Communication Acts including: — post of offensive information on social network sites — send offensive, indecent or threatening messages on a public electronic communications network — use networks without permission
Environmental impact	♦ Description and implications of the impact of the energy use of computer systems and how it could be reduced including: — settings on monitors — power down settings — leaving computers on stand-by ♦ Description and exemplification of the correct ways to dispose of IT equipment including: — recycle individual components appropriately — extraction of dangerous elements — re-use of systems for other uses

Component 2 — assignment
The purpose of the assignment is to assess practical application of knowledge and skills from across the Course to develop a solution to an appropriately challenging computing science problem. It will assess learners' skills in analysing a problem, designing, implementing and testing a solution to the problem, and reporting on that solution. The assignment Component of the Course assessment will require learners to apply knowledge and understanding of a sample of the topics listed in **both tables** above.

Administrative information

Published: April 2016 (version 1.3)

History of changes to Course Assessment Specification

Version	Description of change	Authorised by	Date
1.1	Further information and clarification on scope and structure of the question paper and assignment given in the 'Structure and coverage of Course assessment' section; 'Further mandatory information' section restructured and further information added.	Qualifications Development Manager	June 2013
1.2	'Standardised pseudocode' changed to 'SQA standardised reference language' and 'Contemporary developments' has been added to the requirements for Course coverage in Appendix 2.	Qualifications Manager	June 2015
1.3	Reference to the Question Paper Brief added to the 'Structure and coverage of the Course assessment' section. Depth and clarification added to the mandatory course content.	Qualifications Manager	April 2016