NQ Verification 2017–18
Key Messages Round 2

01 Section 1: Verification group information

Verification group name:	**Computing Science**
Verification event/visiting information	**Event**
Date published:	**June 2018**

National Courses/Units verified:

C716 76	Higher	Assignment (IACCA*)
C716 77	Advanced Higher	Project (IACCA*)

*Internally-assessed component of course assessment

02 Section 2: Comments on assessment

Assessment approaches

All centres had used an SQA produced assignment; however, this was not always the latest version. Centres should ensure that they are using the latest version of the assignment.

Assessment judgements

Advanced Higher Project (IACCA)

Assessors should ensure that candidates select projects that meet the criteria of an Advanced Higher project and that they are capable of implementing. Many candidates were overly ambitious and as a result could not complete projects to the required standard. Candidates should use the Project ideas checklist in the Instructions and guidance for candidates to ensure that their chosen project meets the basic requirements before they embark on the project.

Many projects had elements that were at Higher level. For example, test plans that only include normal, extreme, and exceptional testing; evaluations that only mention evaluation criteria (for example robust and reliable) from the Higher

course. Candidates should be encouraged to look at the Advanced Higher course specification and to ensure that their project addresses the requirements of the Advanced Higher course.

Reflective comments are required on every section. They can be detailed within the section or in the Record of Progress. They should, however, be distinct from the evaluation. Multiple versions of a document shows iteration, but meaningful comments on why changes/decisions were made must be included in order to make them reflective.

Project plan

Many candidates had little evidence that user surveys had been completed or that an analysis of findings had taken place. A blank survey is not sufficient evidence for this.

Requirements specification

Requirements stated at this stage are integral to the rest of the project and should be referenced at every other stage.

Test plan

Clear evidence of a structured test plan that will be used in final testing should be produced. At this stage it should not show final testing.

Interface design

Wireframes should show an outline of the interface and should not be fully illustrated screens created within the implementation software.

Program/data structure design

Pseudocode should be numbered and show main steps and refinements. This should always be done in advance of implementation, not retrospectively. Many candidates had clearly copied and pasted from their implemented code.

Other design notations can be used, for example data flow diagrams, flow charts, or structure diagrams.

When databases or websites are being implemented, the data dictionary, ERDs and query designs should be included.

Implementation

Candidates must implement the project using two Advanced Higher techniques. If not, the candidate will not have:

- a program/data structure that matches the design
- a program/data structure that meets the requirements
- an implementation that reflects the appropriate range of techniques

In this situation, more than half of the total marks become inaccessible to the candidate.

It is helpful for internal and external verifiers if the assessor annotates the candidate's code to showing where the two techniques have been applied.

Final testing

This should be evidenced by screenshots of a working program, not just statements to the effect that 'everything works', which are not sufficient evidence.

Evaluation

The Record of Progress is a separate document from the evaluation but it should be used to inform the evaluation. The evaluation should be evidenced against the evaluation stated on pages 39-40 of the coursework assessment task, and not using Higher evaluation criteria.

Higher Assignment (IACCA)

Stage 2 Building a solution (modular program design)

The purpose of data flow is to show, at the design stage, which variables are being passed in or out of procedures. The programming language that will be used is irrelevant at this stage. Many assessors were accepting IN/OUT as being acceptable for all variables in every procedure.

The design for the user interface should show the inputs and outputs of the program.

The design for importing and exporting data from an external file should show that it is opening and closing the file. Candidates should not use language-specific statements in the design stage.

Stage 2 Building a solution (modular program development)

Candidates should ensure they provide evidence of all their testing.

Stage 2 Building a solution (information system design)

Candidates should show all the validation information in the data dictionary, for example range check >=0 AND <=10, restricted choice S or W.

Stage 2 Building a solution (information system development)

Many candidates did not provide evidence of all the requirements for this stage. Assessors should remind candidates to use the candidate checklist to help ensure that all required evidence is provided.

03 Section 3: General comments

The Assessor's commentary on the candidate marking sheet is essential**.** Without the commentary it is not possible for the internal or external verifier to understand the assessor's judgements. Assessors can also add comments on candidates' completed assignments to explain how they arrived at their decisions on the appropriate banding. This not only helps the assessor come to their decision, but is helpful to both the internal and external verifier.

Where there is disagreement between an assessor and internal verifier, it is necessary to provide evidence of dialogue between assessor and internal verifier, including clear final marks and why those marks were reached.