Course report 2023

Higher Computing Science

This report provides information on candidates' performance. Teachers, lecturers and assessors may find it useful when preparing candidates for future assessment. The report is intended to be constructive and informative, and to promote better understanding. You should read the report in conjunction with the published assessment documents and marking instructions.

The statistics in the report were compiled before any appeals were completed.

Grade boundary and statistical information

Statistical information: update on courses

Number of resulted entries in 2022: 3,491

Number of resulted entries in 2023: 3,562

Statistical information: performance of candidates

Distribution of course awards including minimum mark to achieve each grade

A	Number of candidates	1,294	Percentage	36.3	Cumulative percentage	36.3	Minimum mark required	83
B	Number of candidates	633	Percentage	17.8	Cumulative percentage	54.1	Minimum mark required	70
C	Number of candidates	557	Percentage	15.6	Cumulative percentage	69.7	Minimum mark required	58
D	Number of candidates	511	Percentage	14.3	Cumulative percentage	84.1	Minimum mark required	45
No award	Number of candidates	567	Percentage	15.9	Cumulative percentage	100	Minimum mark required	N/A

Please note that rounding has not been applied to these statistics.

You can read the general commentary on grade boundaries in the appendix.

In this report:

- ‘most’ means greater than 70%
- ‘many’ means 50% to 69%
- ‘some’ means 25% to 49%
- ‘a few’ means less than 25%

You can find more statistical reports on the statistics and information page of SQA’s website.

Section 1: comments on the assessment

Question paper

In the question paper, 58% of candidates completed the ‘Database design and development’ section and 42% completed the ‘Web design and development’ section.

Most questions performed as expected, with grade boundaries moving significantly closer to notional. This reflects a greater understanding by candidates and centres of the standard and level of demand of this course.

However, feedback and analysis on statistical data indicated that several items proved more demanding than intended. Question 9(d)(iii), which was intended to be a ‘C’-level question, proved more demanding, with candidates being asked to relate local variables to maintainability. Question 10(a) was an ‘A’ discriminator, but still resulted in a level of demand that warranted some adjustment to grade boundaries. In question 17(a), very few candidates could identify a functional requirement using the correct technical language, for example a form element or an element to watch or upload videos. While these were all valid questions, candidates’ level of engagement with them was considered when setting grade boundaries.

Assignment

In the assignment, 57% of candidates completed the ‘Database design and development’ section and 43% completed the ‘Web design and development’ section.

The assignment performed in line with expectations.

Feedback from markers, teachers and lecturers indicates that the assignment was positively received, fair and accessible for candidates. Most candidates demonstrated competence in the practical application of the course specification and completed two tasks in the allocated time.

No changes were made to grade boundaries in relation to the assignment.

Section 2: comments on candidate performance

Areas that candidates performed well in

Question paper

Software design and development, and computer systems

Question 1: Many candidates were competent in the conversion from 8-bit two's complement into denary.

Question 3(a): Many candidates could confidently achieve 2 of the 3 marks, with the exponent mark continuing to be the most challenging.

Question 4: Many candidates could state the purpose of a public key.

Question 9(b): Identification of formal parameters has improved when compared to previous exam diets.

Question 9(d)(i): Most candidates were able to identify a local variable, with some candidates correctly identifying the formal parameter as a possible response.

Question 11(a): Most candidates were familiar with the definition of a record data structure and the declaration of a 1D array based on that structure, achieving 3 of the 4 marks.

Database design and development

Question 12: Most candidates were proficient in the drawing of an entity-relationship diagram.

Question 13: Many candidates were able to state the expected output for a given SQL SELECT statement.

Question 15(b): Many candidates could use an aggregate function in conjunction with two tables, and the inclusion of the correct field to GROUP BY.

Web design and development

Question 17(b): Most candidates could draw the navigational structure of a website for the given scenario.

Question 18(a): Most candidates identified the need for and correctly applied a grouping selector.

Question 18(b): Many candidates correctly used float: left or display: inline.

Question 19(a): Most candidates could design a form for the given scenario, with a few candidates still omitting to include a submit button.

Question 19(c): Most candidates identified browser and device compatibility.

Areas that candidates found demanding

Question paper

Software design and development, and computer systems

Question 8(b): Many candidates did not reference the code to the extent required, incorrectly identifying that the ELSE would always be triggered (which is not true if an IF condition matched) instead of noting that the ELSE is triggered even when the number does not match.

Question 9(d)(iii): Most candidates did not make the link between local variables and maintainability and did not answer in the context of the question.

Question 10(a): Most candidates could not identify a boundary from the information in the stem of the question. Instead, candidates either restated the stem or identified a consequence, for example division by 11 will result in a remainder in the range 0 to 10.

Database design and development

Question 15(d): Some candidates identified 'uniqueness' but many did not include the detail that the primary or compound key needs to be unique.

Web design and development

Question 17(a): Most candidates were not able to reference functional requirements using the correct technical language. Instead of identifying the use of forms or elements, they simply restated a part of the stem.

Question 19(b)(i): Although some candidates identified that the code meant that there would not be bullet points on the navigation bar, most candidates did not identify that the li a {… color: grey…} would result in grey text on a grey background.

Areas that candidates performed well in or found demanding

Assignment

Software design and development

Task 1(b): Many candidates were not able to correctly indicate the parameters that are passed in or out of a sub-program. Many candidates failed to identify array and non-array variables. Some candidates did not attempt this question.

Task 1(c): Most candidates were able to implement a modular program with appropriate procedures and standard algorithms. Most candidates followed the supplied program design and made correct use of five parallel arrays with appropriate parameter passing.

Task 1(e): Many candidates did not pay close attention to the wording of the question, which specifically referred to the first sub-program and asked for reference to data structures and loops. Candidates offered generic National 5 standard responses referring to internal commentary, white space and meaningful variable names, none of which were appropriate answers for this question.

Database design and development

Task 2(b): Most candidates were able to implement the SQL statement requiring an aggregate function, equijoin, wildcard, GROUP BY and ORDER BY.

Task 2(c): Most candidates were able to implement the SQL statement or statements requiring an aggregate function, equijoin, complex condition and two queries.

Task 2(d): Most candidates were able to re-write the SQL statement to include the SUM aggregate function and a GROUP BY.

Web design and development

Task 3(a): Many candidates did not indicate a feature of the software when identifying functional requirements.

Task 3(b): Most candidates were able to complete the wireframe, including all required elements, validation and a submit button.

Task 3(c): Most candidates were able to implement HTML, CSS and JavaScript to hide and display sections.

Task 3(d): Most candidates were able to implement the final design, including side-by-side display and appropriate validation.

Task 3(e): Many candidates did not pay close attention to the wording of the question, which specifically referred to two different types of form elements. Elements should be fully tested to establish what will work and what will not work. More than one test per element is required to meet the criteria of 'fully tested'. Candidates may mention extreme, normal and exceptional data; however, this must be supported with data examples relevant to the context of the question.

Task 3(f): Most candidates were able to review the website and comment on fitness for purpose.

Section 3: preparing candidates for future assessment

There has been a significant improvement in candidate performance compared to previous years, with grade boundaries moving much closer to notional. A greater number of exam and assignment past papers means that candidates, teachers and lecturers are more familiar with the standard required. There has been improvement in identification of formal and actual parameters, design of algorithms, the implementation of code, and the use of trace tables to identify errors.

While most teachers and lecturers continue to deliver the course content detailed in the Higher Computing Science Course Specification, some deviate from this content when teaching practical implementation. The course specification exists to ensure consistent and transparent assessment year-on-year. Marking instructions are designed to assess the course content. Candidates are at risk of not being able to access all available marks for a question or task if they use techniques or constructs that are not specified in the course specification.

All standard algorithms should be implemented as refined steps of code and should not use inbuilt features of the software. Teachers and lecturers should adhere to the list of SQL operations, HTML, CSS and JavaScript code provided.

An area of relatively poor performance continues to be within the analysis stage in both the core and optional topics. Candidates attempting the ‘Database design and development’ option showed improvement in identifying functional requirements, however, candidates attempting the ‘Web design and development’ option did not always include a feature of the software (for example ‘a form to … ‘, ‘a button to … ‘).

Improvement is required in the following areas:

- Boundaries need to be elicited from the scenario given, and not simply a restatement of the text given in the scenario.
- Processes in software design and development cannot be inputs or outputs and should identify a manipulation of the data.
- Data flow in software design and development should use brackets for 1D arrays and not use them for single variables.
- Functional requirements in database design and development should identify the underlying code, for example queries, type of query, aggregate functions or other calculations for the scenario given.
- Functional requirements in web design and development should identify the underlying code, for example use of form or other elements and JavaScript.

Centres should ensure candidates appreciate the value of the analysis, design, testing and evaluation stages of the development process. Candidates should complete these sections in the context of the question or task and to a standard that is appropriate for Higher level. Written explanations must be in context and should not be generic, particularly in the analysis or evaluation stages.

All candidates should be encouraged to attempt large tariff questions or tasks, as these are typically constructed to include ‘C’-level marks as well as ‘A’ discriminator marks.

For the assignment, all printed evidence should be legible. Candidates should be encouraged to use the checklist provided to ensure that all required evidence is submitted.

In software design and development, centres should remind candidates to follow the top-level design provided when implementing the task. Internal commentary should be in the context of the program being developed and, while essential, is not required for every line of code. Data flow design continues to be an area that candidates find demanding, and teachers and lecturers should increase classroom practice of this section of the course to improve candidate understanding.

Centres must consider the prior attainment and viability of candidates who are entered in the course, as there is a contingent of candidates who cannot access the Higher level of demand.

Appendix: general commentary on grade boundaries

SQA's main aim when setting grade boundaries is to be fair to candidates across all subjects and levels and maintain comparable standards across the years, even as arrangements evolve and change.

For most National Courses, SQA aims to set examinations and other external assessments and create marking instructions that allow:

- a competent candidate to score a minimum of 50% of the available marks (the notional grade C boundary)
- a well-prepared, very competent candidate to score at least 70% of the available marks (the notional grade A boundary)

It is very challenging to get the standard on target every year, in every subject at every level. Therefore, SQA holds a grade boundary meeting for each course to bring together all the information available (statistical and qualitative) and to make final decisions on grade boundaries based on this information. Members of SQA's Executive Management Team normally chair these meetings.

Principal assessors utilise their subject expertise to evaluate the performance of the assessment and propose suitable grade boundaries based on the full range of evidence. SQA can adjust the grade boundaries as a result of the discussion at these meetings. This allows the pass rate to be unaffected in circumstances where there is evidence that the question paper or other assessment has been more, or less, difficult than usual.

- The grade boundaries can be adjusted downwards if there is evidence that the question paper or other assessment has been more difficult than usual.
- The grade boundaries can be adjusted upwards if there is evidence that the question paper or other assessment has been less difficult than usual.
- Where levels of difficulty are comparable to previous years, similar grade boundaries are maintained.

Grade boundaries from question papers in the same subject at the same level tend to be marginally different year on year. This is because the specific questions, and the mix of questions, are different and this has an impact on candidate performance.

This year, a package of support measures was developed to support learners and centres. This included modifications to course assessment, retained from the 2021–22 session. This support was designed to address the ongoing disruption to learning and teaching that young people have experienced as a result of the COVID-19 pandemic while recognising a lessening of the impact of disruption to learning and teaching as a result of the pandemic. The revision support that was available for the 2021–22 session was not offered to learners in 2022–23.

In addition, SQA adopted a sensitive approach to grading for National 5, Higher and Advanced Higher courses, to help ensure fairness for candidates while maintaining

standards. This is in recognition of the fact that those preparing for and sitting exams continue to do so in different circumstances from those who sat exams in 2019 and 2022.

The key difference this year is that decisions about where the grade boundaries have been set have also been influenced, where necessary and where appropriate, by the unique circumstances in 2023 and the ongoing impact the disruption from the pandemic has had on learners. On a course-by-course basis, SQA has determined grade boundaries in a way that is fair to candidates, taking into account how the assessment (exams and coursework) has functioned and the impact of assessment modifications and the removal of revision support.

The grade boundaries used in 2023 relate to the specific experience of this year's cohort and should not be used by centres if these assessments are used in the future for exam preparation.

For full details of the approach please refer to the National Qualifications 2023 Awarding — Methodology Report.

