Next Generation Higher National Educator Guide

Higher National Diploma Computer Science

Qualification code: GV20 48

This qualification is available in a restricted delivery model from academic session 2025

This guide provides information about the Higher National Diploma (HND) to ensure consistent and transparent assessment year on year. It is for lecturers and assessors, and contains all the mandatory information you need to deliver and assess the HND.

You must read it alongside the Grading Pack.

Published: September 2025 (version 1.0)

Contents

Purpose of the qualification **1**
Structure **2**
Framework 3
Aims of the qualification **4**
General aims 5
Specific aims 5
Who is this qualification for? 6
Recognising prior learning 6
Articulation and progression 7
Credit transfer arrangements 7
Recommended Core Skills entry profile 7
How the qualification meets employer and higher education institution needs **8**
Table 1: mapping qualification aims to units 9
Table 2: mapping National Occupational Standards (NOS) and/or trade or professional body requirements to units 12
Table 3: assessment strategy for the qualification 15
Meta-skills **18**
Meta-skills in HND Computer Science 19
Learning for Sustainability **26**
Context 26
Learning for Sustainability in Next Generation Higher National Qualifications 27
Grading **30**
Whole-qualification grade outcomes 30
Whole-qualification grade descriptors 31
Approaches to delivery and assessment **32**
Sequencing or integrating units 32
Additional guidance on integrated or holistic assessment 33
Remediation and re-assessment in Next Generation Higher National Qualifications 37
Information for centres **39**
Equality and inclusion 39
Internal and external verification 39

Glossary **40**
Information for learners **41**
HND Computer Science 41
Qualification information 41
Administrative information **44**
History of changes 44

Purpose of the qualification

Higher National Diploma (HND) Computer Science provides learners with the high-quality technical and vocational skills they need to meet the rapidly changing demands of modern computing and IT technologies. There is an increasing demand for computer science professionals to sustain and develop the applications of computing technologies in modern society, business and industry. While there are many specialisms in the field of computing, there are fundamental concepts and technologies that underpin them all.

Computing and its applications are an ever-evolving field, with aspects such as artificial intelligence, cyber security and cloud technologies increasing in importance. Learners will gain knowledge and understanding of current technologies and applications of computer science. They work collaboratively in their learning.

The qualification prepares learners for a career path in this field. It develops foundational skills and knowledge in important aspects of computer science, including programming, data structures, algorithms, and computer systems. Learners can specialise in a particular area of computer science such as artificial intelligence (AI), cyber security or data science.

The qualification offers opportunities for learners to develop the meta-skills and professional attitudes required in this vital industry sector, along with awareness of legal and ethical obligations, including those relating to diversity, inclusion and sustainability. It also prepares learners for further study in this specialism, or in other aspects of computing, including study at degree level.

The structure and aims of the HND are described in the following sections.

Structure

Higher National Diplomas (HNDs) are at SCQF level 8 and are made up of 120 SCQF credit points (15 SQA credits). HNDs must incorporate at least 80 credit points (10 SQA credits) at SCQF level 8.

HND Computer Science meets the Next Generation Higher National (NextGen: HN) design principles, by incorporating a mandatory project unit (Professional Practice in Computer Science at SCQF level 8) with an SQA credit value of 4.

The tables that follow show how the mandatory and optional units contribute to the general and specific aims of the HND.

There are two mandatory units in the structure that contribute 7 SQA credits at SCQF level 8. The first of these (Computer Science) deals with the concepts and principles that are fundamental to the practice of computer science. The other mandatory unit is a project unit that requires group working. Project work for this unit can include research into a topic in computer science, such as programming languages or developing a computing solution to a problem from contexts such as science, engineering, health, business or cybersecurity.

The options and restricted options groups enable learners to develop an area of specialism, such as programming paradigms or AI. Learners accumulate the remaining 8 SQA credits from a selection of optional units from the framework.

Framework

The HND is made up of mandatory and optional units. Learners must complete all the mandatory units and 8 SQA credits from the optional units.

Mandatory units

Unit code	Unit title	SQA credits	SCQF credit points	SCQF level
J68M 48	Computer Science	3	24	8
J7DX 48	Professional Practice in Computer Science	4	32	8

Optional units

Unit code	Unit title	SQA credits	SCQF credit points	SCQF level
J7DH 48	Algorithms and Data Structures	3	24	8
J7DT 48	Applied Mathematics in Computing	2	16	8
J7DJ 48	Applied Artificial Intelligence	3	24	8
J7DV 48	Database Design and Development	3	24	8
J691 47	Emerging Technologies and Experiences	1	8	7
J7DW 48	Object Oriented Programming	2	16	8
J7DY 48	Programming Paradigms	2	16	8
J7E0 48	Virtualisation Technologies	2	16	8

Restricted optional units (maximum of 3 credits)

Unit code	Unit title	SQA credits	SCQF credit points	SCQF level
J68T 48	Networking and Infrastructure	3	24	8
J7E1 48	Application Development for Web	3	24	8

Learners must pass the two mandatory units (7 SQA credits) and achieve at least 8 SQA credits from the optional units specified in the framework. No more than 3 SQA credits can be obtained from the units in the restricted optional group.

We have provided information relating to the grade of the award later in this Educator Guide.

Aims of the qualification

HND Computer Science develops learners' knowledge and understanding of the fundamental concepts and principles of computer science, such as algorithms, data structures and programming languages. It enables them to write efficient and maintainable code and solve real-world problems that require computer solutions.

The HND includes an extensive collaborative project for learners to design and develop a solution to a computing problem using project management and other formal development processes, including version control, testing, and technical documentation.

General aims

1. develop a range of academic competencies at SCQF level 8, including study and research skills
2. develop vocational skills to prepare learners for employment in current and future roles
3. develop capacity to apply knowledge and skills to solve real-world problems
4. develop meta-skills that complement technical and professional knowledge and skills
5. develop Learning for Sustainability skills, knowledge, understanding and values

Specific aims

1. develop understanding of the key concepts and principles of computer science
2. develop understanding of current and emerging technologies in computer science, including artificial intelligence and cyber security
3. develop mathematical and statistical understanding as it relates to computer science
4. prepare learners for employment in the design, development, testing and implementation of computing solutions
5. select and apply the correct approaches to resolving problems through computing solutions
6. develop learners' computational thinking, pattern recognition, deconstruction, logical thinking, synthesis and analysis skills
7. help learners to gain understanding of the ethical, social and legal issues associated with the use of computer systems, including those relating to diversity, inclusion and sustainability
8. prepare learners for progression to further studies in computer science, or related disciplines, at SCQF level 9

Who is this qualification for?

This qualification is suitable for learners who wish to develop their knowledge and understanding of fundamental concepts in computer science. It provides them with a sound foundation for further study of aspects of computer science that are central to the development of modern solutions to computing problems. To benefit from this qualification, learners should understand the basics of how computers work and be familiar with computer program design, coding and testing.

Entry to this qualification is at your centre's discretion. However, we recommend that learners have one or more of the following:

- HNC Computing (NextGen) GT6G 47
- HNC Computing GF3E 15
- equivalent qualifications and/or experience

Learners can progress to:

- other qualifications in computing-related areas
- degree-level study in computer science
- further study, employment and/or training in computing-related areas

Recognising prior learning

SQA recognises that learners gain knowledge and skills through formal, non-formal and informal learning contexts. Formal learning is learning certificated by a recognised awarding or professional body. Non-formal learning includes learning such as employers' in-house training courses. Informal learning is learning based on experience from a variety of environments that is not formally assessed.

It is unlikely that a learner would have the appropriate prior learning and experience to meet all the requirements of a full HND.

You can find more information and guidance about the recognition of prior learning on SQA's website.

Articulation and progression

Learners who complete this qualification could go on to:

- other qualifications in computing or related areas
- further study, employment and/or training

HND Computer Science is not designed to articulate to any particular degree programme. However, the design process has taken account of the common themes that occur in sub-degree courses in Computer Science at Scottish universities to facilitate progression to degree level in such courses. Centres can offer the selection of optional units that best fits progression to a chosen university degree course.

Credit transfer arrangements

Centres can make decisions about transferring credit. They can transfer credit if the subject-related content of the units is broadly equivalent. Centres should consider the currency of a learner's achievement before transferring credit.

Recommended Core Skills entry profile

Learners should have the following Core Skills at the stated SCQF levels before starting this qualification. This information can help identify learners who may need additional support.

Core Skill	Recommended SCQF entry profile
Communication	SCQF level 6
Numeracy	SCQF level 6
Information and Communication Technology (ICT)	SCQF level 6
Problem Solving	SCQF level 6
Working with Others	SCQF level 6

How the qualification meets employer and higher education institution needs

This qualification is designed in collaboration with employers, higher education institutions (HEIs), practitioners and professional bodies to meet the sector need.

The following tables show how the qualification can benefit employers and HEIs by equipping learners with the necessary skill set:

- Table 1 shows how units map to the aims of the qualification.
- Table 2 shows how the units map to National Occupational Standards (NOS)
- Table 3 shows the assessment strategy for the qualification.

Table 1: mapping qualification aims to units

General aims

Key: aim is directly relevant to unit (X), aim is optional in this unit (O), aim is not applicable to this unit (N/A)

Unit code	Unit title	General aim 1	General aim 2	General aim 3	General aim 4	General aim 5
J68M 48	Computer Science	X	X	X	X	X
J7DX 48	Professional Practice in Computer Science	X	X	X	X	X
J7DH 48	Algorithms and Data Structures	X	X	X	X	X
J7DT 48	Applied Mathematics in Computing	X	—	X	X	X
J7DJ 48	Applied Artificial Intelligence	X	X	X	X	X
J7DV 48	Database Design and Development	X	X	X	X	X
J691 47	Emerging Technologies and Experiences	X	—	—	X	X

Unit code	Unit title	General aim 1	General aim 2	General aim 3	General aim 4	General aim 5
J7DW 48	Object Oriented Programming	X	X	X	X	X
J7DY 48	Programming Paradigms	X	X	X	X	X
J7E0 48	Virtualisation Technologies	X	X	X	X	X
J68T 48	Networking and Infrastructure	X	X	X	X	X
J7E1 48	Application Development for Web	X	X	X	X	X

Specific aims

Unit code	Unit title	Specific aim 1	Specific aim 2	Specific aim 3	Specific aim 4	Specific aim 5	Specific aim 6	Specific aim 7	Specific aim 8
J68M 48	Computer Science	X	X	—	X	X	X	X	X
J7DX 48	Professional Practice in Computer Science	X	X	—	X	X	X	X	X
J7DH 48	Algorithms and Data Structures	X	—	X	X	X	X	—	X

Unit code	Unit title	Specific aim 1	Specific aim 2	Specific aim 3	Specific aim 4	Specific aim 5	Specific aim 6	Specific aim 7	Specific aim 8
J7DT 48	Applied Mathematics in Computing	X	—	X	—	—	X	—	X
J7DJ 48	Applied Artificial Intelligence	X	X	—	X	X	X	X	X
J7DV 48	Database Design and Development	X	X	—	X	X	X	X	X
J691 47	Emerging Technologies and Experiences	X	X	—	—	—	—	X	X
J7DW 48	Object Oriented Programming	X	X	—	X	X	X	—	X
J7DY 48	Programming Paradigms	X	X	—	X	X	X	—	X
J7E0 48	Virtualisation Technologies	X	X	—	X	X	X	X	X
J68T 48	Network Infrastructure	X	X	—	X	X	X	X	X
J7E1 48	Application Development for Web	X	X	—	X	X	X	X	X

Table 2: mapping National Occupational Standards (NOS) to units

The NOS for Information Technology Professionals are organised into 11 disciplines, each with a set of sub-disciplines. The following information lists the sub-disciplines that cover the key areas in HND Computer Science:

1 Digital Leadership Personal Competencies

- 1.1 Business Competencies

2 Business Analysis and Change Management

- 2.6 Requirements Engineering

4 Architecture, Analysis and Design

- 4.1 Systems Architecture
- 4.2 Data Analysis
- 4.4 Systems Analysis
- 4.5 Data Design
- 4.8 IT Infrastructure Design and Planning

5 Solution Development Systems Development

5.1 Software Development

5.5 Software Process Improvement

5.6 DevOps

5.7 User Centred Development

6 Cyber Security (IT Professional)

6.1 Information Security Management

7 Service Management and Delivery

7.1 IT Service Operations and Event

7.8 Change and Release Management

8 Data Science

8.3 Data Science

8.4 Artificial Intelligence (AI)

8.5 Machine Learning

9 IT Networks Network Services Planning

9.1 Network Design

Unit code	Unit title	NOS codes
J68M 48	Computer Science	4.1, 4.2, 5.1, 6.1, 9.1
J7DX 48	Professional Practice in Computer Science	1.1, 2.6, 4.1, 7.1, 8.1, 9.1
J7DH 48	Algorithms and Data Structures	5.1, 5.2, 5,5, 5.6, 8.3, 8.4, 8.5
J7DV 48	Database Design and Development	4.1, 4.2, 4.5, 8.1
J7DW 48	Object Oriented Programming	5.1, 5.7
J7DY48	Programming Paradigms	5.1
J7E0 48	Virtualisation Technologies	4.1, 4.8, 7.1, 9.1
J7DJ 48	Applied Artificial Intelligence	4.2, 4.5, 6.1, 8.3, 8.4, 8.5
J691 47	Emerging Technologies and Experiences	4.1, 5.1, 6.1, 7.1, 8.1, 9.1
J68T 48	Network Infrastructure	2.6, 4.1, 4.8, 7.1, 9.1
J7E1 48	Application Development for Web	1.1, 5.1, 5.2, 5.7, 8.1

Table 3: assessment strategy for the qualification

Unit code	Unit title	Assessment method
J68M 48	Computer Science	Knowledge evidence: No sampling frame is provided for question paper. Otherwise, there should be at least one evaluation of a contemporary ethical issue in computer science. Product evidence: One or more assembly language programs. One or more relational (normalised) databases.
J7DX 48	Professional Practice in Computer Science	Product evidence (team): This is an extended team project of sufficient size and complexity to require a team solution. The project team will produce a report of how they planned and managed the project, analysed requirements, designed and implemented a solution, and their conclusions and recommendations. The project may also be a research question to be addressed, in which case the report will relate to the design and conduct of the research and the conclusions and/or recommendations from it. Product evidence (individual): A reflective report on the individual learner's contribution to the project. Performance evidence: A team presentation of the conduct of the project, the outcomes and solution. Meta-skills: A reflective report showing that the individual has self-assessed their meta-skills, created a plan for development and carried out activities that improved and developed their meta-skills. Learning for Sustainability: A report on how a computer science process or product could be made more sustainable to meet the aims of at least two selected UN SDGs.

Unit code	Unit title	Assessment method
J7DH 48	Algorithms and Data Structures	Product evidence: Learners design, code and test one or more computer programs that cover all skills components. The evidence requirement specifies a list of mandatory functions to be coded in lightly-controlled conditions.
J7DT 48	Applied Mathematics in Computing	Knowledge evidence: A question paper with a sampling frame provided. Product evidence: Learners use software or write program code to carry out mathematical and statistical operations to satisfy a list of requirements.
J7DJ 48	Applied Artificial Intelligence	Knowledge evidence: Sampling frame provided for question paper. Otherwise, learners can develop a portfolio of research or investigations. Product evidence: From a given brief, learners select and use AI services or tools and interpret the results.
J7DV 48	Database Design and Development	Product evidence: From a given brief, learners design and develop a normalised relational database and populate it. They create a set of queries to access, manage and analyse the data. From a given brief, learners design and develop a NoSQL database. They create a set of queries to manage and analyse NoSQL data.
J691 47	Emerging Technologies and Experiences	Knowledge evidence: In any form of media, learners demonstrate knowledge relating to each outcome.
J7DW 48	Object Oriented Programming	Product evidence: Learners investigate and apply OOP techniques. They implement a given object-oriented design, produce test documentation, and troubleshoot code errors.

Unit code	Unit title	Assessment method
J7DY 48	Programming Paradigms	Knowledge evidence: Learners explain and give examples of each paradigm. They compare and contrast paradigms, taking account of key characteristics and use cases. Where a question paper is used, a sampling frame is suggested. Product evidence: Learners solve a given problem using at least two of the paradigms and produce an evaluation report on the outcomes.
J7E0 48	Virtualisation Technologies	Knowledge evidence: Sampling frame (extensive) provided for question paper. Product evidence: Learners demonstrate their competence in building a virtual machine and deploying applications to a virtual machine in the cloud. They demonstrate virtualisation on desktop PC.
J68T 48	Networking and Infrastructure	Knowledge evidence: A theoretical assessment. Can be a closed-book test. Product evidence: From a given scenario, learners design and build a network, including cabling and configuration of network services. Learners demonstrate their practical skills in network troubleshooting and testing.
J7E1 48	Application Development for Web	Product evidence: From a given brief, learners design and build a secure, full-stack, interactive web app using current front-end and back-end technologies. They deploy to the internet, use version control and repositing, and document the app, including testing.

Meta-skills

Every NextGen: HN Qualification gives learners the opportunity to develop meta-skills.

Meta-skills are transferable behaviours and abilities that help people to adapt and succeed in life, study and work. There are three categories of meta-skills: self-management, social intelligence and innovation. Each of these is made up of four meta-skills and a number of sub-skills.

- Self-management — focusing, integrity, adapting, initiative
- Social intelligence — communicating, feeling, collaborating, leading
- Innovation — curiosity, creativity, sense-making, critical thinking

From early in the qualification, we want learners to identify and understand the meta-skills they can develop, and to appreciate the personal and professional value of these skills. We want to support learners to continue to articulate, use and build on them long after they have achieved their qualification. In this way, we help learners to develop broad skills profiles, enabling them to thrive in a changing world.

Every NextGen: HN unit signposts opportunities for learners to develop meta-skills, and there is an assessed outcome in one of the mandatory units. When you make your whole-qualification grade decisions, you consider learners' commitment to engaging with meta-skills development.

You do not assess learners on their competence or progress in individual meta-skills. Instead, you assess them on evidence that they have engaged with a personal process of development. Meta-skills development is founded on a clear process of self-assessment, goal setting, action planning and reflective practice.

You can find meta-skills teaching, learning and assessment resources on SQA's meta-skills web page.

Meta-skills in HND Computer Science

You can introduce meta-skills to learners as tools they can use in response to real-world challenges and opportunities. At SCQF level 8, you should use terminology from the Skills 4.0 model, but it is important that you develop a shared understanding with learners about meta-skills and what they mean to them, both individually and in the context of coursework, projects and sectors.

You should embed meta-skills in learning and project tasks as a context for planning, practice, and reflection. You should encourage learners to be self-aware, set active goals and monitor their progress.

The process of developing meta-skills is not linear and you should ensure learners are active participants in their learning. At the start of the process, you should introduce meta-skills to learners and explore the concept of self-assessment with them. You should set goals and make development and evaluation plans together. The process should become cyclical, with reflective practice informing new self-awareness, goal-setting and review.

Many traditional learning and teaching activities used to develop industry or sector-specific skills, knowledge and understanding also support the development of meta-skills. You can map these in course materials and resources and during learning.

Meta-skills are central to successfully engaging with and completing assignments and projects. You should encourage learners to plan how they will use and develop meta-skills in their coursework and to reflect on their success and future goals.

The role of the coach, mentor or facilitator is key to help learners understand, develop and reflect on their own meta-skills and those central to course activities, assessment projects and their target industry or sector. You and any employer partners or guest speakers could guide learners by taking on a coaching and mentoring role.

In this role, you should introduce learners to the fundamentals of reflective practice. You could use several models of reflective practice. You do not need to use a

theoretical perspective. Any reference to these models should support learners' understanding of the nature and value of reflective practice in self-understanding and making change.

Introducing reflective practice can support your learners' personal development and goal setting. Frequent formative peer-to-peer, assessor, client (if appropriate) and group reflection activity can support learners through reflective practice.

Learners can focus on any meta-skills appropriate to them and their context. However, learning and teaching should also facilitate individual development. Learners have individual strengths and areas for development, and they do not have to reach a particular level in relation to meta-skills. Coursework and projects provide the context for development appropriate to the SCQF level. Within these contexts, the **process** of development is important. You should create a clear learning plan with each learner to provide evidence of their development.

You can create descriptions of abilities and skills that relate to meta-skills with your learners. These can come from self-profiling, exploring the industry and sector, and discussion with peers and employers. You should consider the meta-skills needed to complete coursework and meet personal goals to set a context for reflection.

Exploring learning and working styles, personality traits and preferences, personal profiling and self-assessment tools can help learners to develop an understanding of their strengths and areas for development.

You can use case studies and scenario-based activities to demonstrate the value of meta-skills and how they can be applied. You can provide opportunities for peer reflection. A group of learners could share experiences and reflections about how to apply meta-skills in the context of their coursework. You could adopt the role of facilitator to draw learners' attention to situations where meta-skills were or could have been applied.

Reflective discussions can focus on how and where meta-skills are being developed. Your discussions with learners could include positive recognition and guidance on future development based on previous performance. As learners progress, you could

introduce industry content that requires skills like problem recognition and problem solving, both of which combine multiple meta-skills.

You can deliver the knowledge and skills for practical aspects of projects in sequence. However, learners benefit from learning and teaching that integrates meta-skills with project planning and development. This approach supports learners to engage in reflective practice throughout the project, and develops their self-awareness and an appreciation for continuous learning. It also maximises your opportunities to support, coach and mentor learners through their projects.

Mapping opportunities to develop meta-skills across HND Computer Science

Self-management

Unit code	Unit title	Meta-skills
J68M 48	Computer Science	Focusing Adapting Initiative
J7DX 48	Professional Practice in Computer Science	Focusing Integrity Adapting Initiative
J7DH 48	Algorithms and Data Structures	Focusing
J7DT 48	Applied Mathematics in Computing	Focusing Initiative
J7DJ 48	Applied Artificial Intelligence	Focusing Integrity Initiative

Unit code	Unit title	Meta-skills
J7DV 48	Database Design and Development	Focusing Integrity Adapting Initiative
J691 47	Emerging Technologies and Experiences	Integrity
J7DW 48	Object Oriented Programming	Focusing
J7DY 48	Programming Paradigms	Focusing Adapting Initiative
J7E0 48	Virtualisation Technologies	Focusing Adapting Initiative
J68T 48	Networking and Infrastructure	Focusing Adapting
J7E1 48	Application Development for Web	Focusing Integrity Adapting Initiative

Social intelligence

Unit code	Unit title	Meta-skills
J68M 48	Computer Science	Communicating Collaborating
J7DX 48	Professional Practice in Computer Science	Communicating Feeling Collaborating Leading
J7DJ 48	Applied Artificial Intelligence	Communicating
J7DV 48	Database Design and Development	Communicating Collaborating Leading
J7DW 48	Object Oriented Programming	Communicating Collaborating
J68T 48	Networking and Infrastructure	Communicating Collaborating Leading
J7E1 48	Application Development for Web	Communicating Collaborating Leading

Innovation

Unit code	Unit title	Meta-skills
J68M 48	Computer Science	Curiosity Sense-making Critical thinking
J7DX 48	Professional Practice in Computer Science	Curiosity Creativity Sense-making Critical thinking
J7DH 48	Algorithms and Data Structures	Creativity Curiosity Sense-making Critical Thinking
J7DT 48	Applied Mathematics in Computing	Sense-making Critical Thinking
J7DJ 48	Applied Artificial Intelligence	Sense-making Critical Thinking
J7DV 48	Database Design and Development	Creativity Sense-making Critical thinking
J691 47	Emerging Technologies and Experiences	Sense-making Critical thinking
J7DW 48	Object Oriented Programming	Creativity Sense-making Critical Thinking
J7DY 48	Programming Paradigms	Creativity Sense-making Critical Thinking

Unit code	Unit title	Meta-skills
J7E0 48	Virtualisation Technologies	Creativity Sense-making Critical thinking
J68T 48	Networking and Infrastructure	Creativity Sense-making Critical thinking
J7E1 48	Application Development for Web	Creativity Critical thinking Sense-making

Learning for Sustainability

Context

The United Nations (UN) 2030 Agenda for Sustainable Development, adopted by the UK in 2015, has shaped the development of Scottish, national and international sustainability policy. It sets out the UN Sustainable Development Goals (SDGs), which are central to the Scottish Government's National Performance Framework. Learning for Sustainability (LfS) is a commitment to embedding the SDGs in Scottish education.

LfS embraces global citizenship, sustainable development, social justice, human rights, climate change, biodiversity loss, equality and inclusion. Learners develop their capacity to deal with the unpredictable social, economic and environmental challenges facing our rapidly changing world.

LfS combines:

- education for sustainable development (ESD)
- global citizenship
- outdoor learning

ESD is the internationally used term for sustainability education. Although LfS has a broader remit, the terms are largely interchangeable. Colleges and universities tend to use ESD, while schools usually use LfS. Both focus on a broad range of social, economic and environmental themes and approaches across all levels of education. SQA uses LfS as an umbrella term.

Learning for Sustainability in Next Generation Higher National Qualifications

Sustainability is a core component in this qualification.

Learners who complete this qualification should have:

- a general understanding of social, economic and environmental sustainability
- a general understanding of the SDGs
- a deeper understanding of subject-specific sustainability
- the confidence to apply the skills, knowledge, understanding and values they develop in the next stage of their life

Sustainability is embedded as an outcome in the Professional Practice in Computer Science project unit.

Learners who complete this outcome can:

- assess their own knowledge and understanding of sustainability and the SDGs
- review unit content against the SDGs to identify a sustainability-related issue
- apply knowledge and understanding of sustainability and the SDGs to propose improvements

You should look for opportunities in the HND Computer Science to demonstrate to your learners the various ways in which they might encounter concepts relating to sustainability. They should understand that computer science can support the realisation of a range of UN SDGs, while in some operational aspects, it also has a negative impact.

Here are some positive contributions that computer science makes to sustainability:

- computer technology can address environmental challenges, such as developing algorithms for optimising energy consumption in buildings or creating smart grids
- sustainable software engineering practices include code optimisation techniques that reduce energy consumption and minimise the carbon footprint of software applications

- data analytics and machine learning contribute to sustainability efforts, such as analysing patterns in energy usage or developing predictive models for environmental conservation
- building applications that facilitate renewable energy integration or developing software solutions for waste management and recycling
- the intersection of computer science with other fields, like environmental science or urban planning, supports the design of innovative and sustainable solutions

The challenges that computer science poses for sustainability include:

- the rapid pace of technology advancement, which often leads to a shorter lifespan for electronic devices and contributes to electronic waste
- high-performance computing and resource-intensive algorithms, which can consume significant amounts of energy and have a negative environmental impact
- privacy and security concerns, which may clash with sustainability goals, as increased data collection and processing for environmental monitoring might raise ethical dilemmas regarding data protection
- the risk of bias and discrimination in algorithmic decision-making systems, where algorithms trained on biased data may perpetuate social and environmental injustices
- the manufacture and disposal of electronic components used in computer science that can generate harmful electronic waste and contribute to the depletion of natural resources

Any of the SDGs can be covered; there are none that are mandatory.

You can cover any of the SDGs that are relevant to the subject area.

Find out more about SQA's approach on the NextGen: HN Learning for Sustainability web page. There is an LfS reflective template available in the resources section. You may find it helpful as a starting point for considering how the SDGs are, or could be, embedded in a qualification, unit or assessment.

Grading

Please see the Grading Pack for this qualification for more information on making grade judgements.

Grading in NextGen: HN Qualifications produces a valid and reliable record of a learner's level of achievement across the breadth of the qualification content.

As well as grading the whole qualification, you assess individual units on a pass or fail basis. Each unit has evidence requirements that learners must achieve before you can consider them for whole-qualification grading.

Whole-qualification grade outcomes

Learners who pass NextGen: HN Qualifications receive one of the following grade outcomes for the qualification as a whole:

- Achieved with Distinction
- Achieved with Merit
- Achieved

To determine a learner's whole-qualification grade, you use the grading matrix provided in the Grading Pack to assess and judge their performance across the key aspects of the HND. You must align your judgements with the following whole-qualification grade descriptors.

Whole-qualification grade descriptors

Achieved with Distinction

The learner has achieved an excellent standard across the course content, going significantly beyond meeting the qualification requirements. They showed a comprehensive knowledge and understanding of course concepts and principles, and consistently used them to apply skills to complete high-quality work. They engaged significantly with the process of developing their meta-skills in the context of their HN Qualification.

Achieved with Merit

The learner has achieved a very good standard across the course content, going beyond meeting the qualification requirements. They showed a very good knowledge and understanding of course concepts and principles, and consistently used them to apply skills to complete work of a standard above that expected for an Achieved grade. They actively engaged with the process of developing their meta-skills in the context of their HN Qualification.

Achieved

The learner has achieved a good standard across the course content, credibly meeting the qualification requirements. They showed a good knowledge and understanding of course concepts and principles, and used them to apply skills to complete work of the required standard. They engaged with the process of developing their meta-skills in the context of their HN Qualification.

Approaches to delivery and assessment

While conventional teaching methods like lectures and discussions are beneficial for introducing foundational knowledge, you should also encourage active participation in discussions, to ensure learners understand the concepts and ideas of computer science. You should also give learners the opportunity to gain presentation skills by assigning research topics to be reported back to the class.

Computer science is a practical field, so it is crucial to engage your learners in labs and workshops where they can apply the theory they have learned. You should set up coding exercises, practical labs on databases, and networking exercises. You can set problems that require learners to apply their knowledge in a practical way, such as designing an algorithm to solve a particular problem.

For some units, you can encourage learners to work collaboratively. The Professional Practice unit is a large extended project in which learners experience more formal approaches to project planning and management. This can mimic a real-world working environment and enhance their teamwork and communication skills. You could also engage learners in collaborative activities, such as code reviews or hackathons.

You should encourage self-guided learning by signposting resources to be accessed outwith class time. Typical sources are industry blogs, podcasts, vendor training videos, and online courses. Inviting industry experts for guest lectures or webinars is another way to expose learners to real-world applications and emerging trends.

Sequencing or integrating units

You should prioritise the Computer Science mandatory unit for delivery as early as possible, as it provides underpinning concepts for many of the optional units. You should also commence the mandatory project unit (Professional Practice in Computer Science) whenever you are confident that the learners have sufficient breadth of knowledge in their optional topics to apply the concepts, principles and skills acquired in them.

You should consider where you can deliver units in tandem to enhance learning, or where sequential delivery might provide the best learning opportunity. In all the units in this HND, the approach you take should be to realise the computer science concepts through practical experiences with real-world problems demanding a computational solution. The more realistic these are, the more powerful the learning experience will be. The increasing availability of virtual cloud infrastructures from technology vendors increases scope to challenge learners to apply their knowledge and skills to unfamiliar contexts.

As indicated above, some units would benefit from co-delivery to enhance the learning experience. Typical examples of these could be:

- Programming Paradigms and Object Oriented Programming
- Algorithms and Data Structures and Applied Mathematics for Computing
- Virtualisation Technologies and Networking and Infrastructure

Many of the units are suited to assessment in the form of assignments and/or projects. In such cases the product evidence is clearly specified in each unit and centres should use this information to plan work that enables the evidence requirements to be met. The case studies and assignments used for the purpose of generating assessment evidence should be as real-world as possible, while considering the SCQF level of the unit. Where program code is required, centres should ensure that learners have sufficient access to programming resources such as an integrated development environment (IDE) or notebook-style coding environments.

Additional guidance on integrated or holistic assessment

Holistic or integrated assessment focuses on assessing a number of outcomes in a unit together, or in some cases, assessing the unit as a whole, rather than by outcome. When assessing a unit of competence holistically, the assessment activities integrate a number of aspects of the competence. Holistic or integrated assessment can reduce the time spent on assessment and can promote greater equity in the assessment process.

When developing or revising a NextGen: HN Qualification, SQA works with a development team to devise an appropriate assessment strategy that accommodates holistic or integrated assessment. However, the practice of integrating units for the purposes of learning and teaching is a centre-led activity.

Units are designed to facilitate holistic or integrated assessment approaches that prevent large, unwieldy assessments.

Sometimes more than one piece of evidence is needed for a unit. For example, if a unit is about creating a code, a learner would need to produce evidence of knowledge (following the software development processes) and product (a working piece of code).

Evidence requirements must do what they say: specify requirements for evidence of learner competence in the unit. The evidence must be of sufficient quality for an assessor or verifier to judge that the learner has achieved the unit.

Assessing project units

The Professional Practice in Computer Science unit comprises a project that is sufficiently large or complex to require analysis and solution by a project team. This could be research into one or more aspects of computer science, such as AI or machine learning. It could also be a problem that requires a computing solution, such as the design and implementation of an algorithm. You group your learners into teams of three to five learners, considering the preferences expressed by learners in relation to the available projects and your judgement of an appropriate composition for each team, which should vary in terms of age, gender, ability and other relevant characteristics.

You should have a range of computer science projects or research themes that would be suitable to enable learners to demonstrate the outcomes of the project unit. The requirements must be sufficiently complex to require formal project management and development approaches. You should select these to be as real-world as possible and aligned to the level of study required. The project should allow learners to draw on the range of outcomes in their Higher National Qualification and make it

possible for them to develop their meta-skills while engaging in the project. To standardise the learner experience and level of demand, you should use a template to describe the objectives and required outcomes for each project to be offered to the groups of learners.

You should:

- approve projects
- assign learners to teams
- ensure each team allocates roles at the beginning of the project
- receive briefings from each team on progress against milestones
- observe and record the contributions of individual learners to teamwork and the problem solution
- arrange for the presentation of the project outcomes and solution
- intervene to guide and support learners when circumstances require it

You should approach your role with a light touch, leaving each team to make their own decisions about a project methodology, as well as roles, timelines, resource allocations and meetings. We recommend that each learner leads a specific part of the project (such as lead designer or lead researcher).

The product evidence is the outcome of the research or the solution to a computer science problem. Each project team must collectively produce items 1 to 5 below. Each learner must individually produce item 6. Learners must provide the following product evidence:

1. the project plan based on requirements gathering and analysis
2. the solution to the problem or the conclusions from the research
3. a project report (including project evaluation)
4. documentation of the process and outcomes
5. presentation of the project outcomes and demonstration of solution
6. personal statement

The learner's personal statement must:

- describe their specific role or roles in the project team
- include a self-evaluation of their contribution to the progress and completion of the project

The performance evidence is in two parts:

- a record of the learner's professional behaviours and contributions to the project; this may be in the form of a checklist
- a recording of performance in delivering the presentation

Each learner must make an identifiable contribution to the final presentation of the project outcomes and the demonstration of the solution (where required). This contribution can be in any form that is appropriate and agreed with the tutor.

Remediation and re-assessment in Next Generation Higher National Qualifications

Remediation

Remediation allows an assessor to clarify learners' responses, either by requiring a written amendment or by oral questioning, where there is a minor shortfall or omission in evidence requirements. In either case, the assessor must formally note such instances, in writing or as a recording, and make them available to the internal and external verifier.

Remediation is not permitted for closed-book assessments.

The size and structure of the larger NextGen: HN units should mean that the assessor or lecturer is close enough to ongoing assessment activity in project-based units to identify the requirement for remediation as it occurs.

Re-assessment

We must give learners who fail the unit a re-assessment opportunity or, in exceptional circumstances, two re-assessment opportunities. Where we have introduced larger units to the framework, we expect instances of re-assessment to be minimal, due to the approach to assessment and remediation. Where re-assessment is required in a project-based unit, a substantially different project must be used.

Resource requirements

To ensure that learners can develop their knowledge and skills across the mandatory and optional units in the framework, you should consider the following resources:

- access to computer workstations that have reliable and fast internet access. These must be sufficient to perform coding exercises, access video-based instruction and support collaboration. Server technology is required for

virtualisation technologies and potentially some AI model training. These can be physical servers or cloud-based servers

- the computer workstations should have good quality graphic processing units (GPUs) for image rendering and for the Artificial Intelligence unit. Networking lab equipment, such as routers, switches and firewalls, are required for the Networking and Infrastructure unit
- software provision should include integrated development environments (IDEs) such as Eclipse, IntelliJ IDEA, or Visual Studio Code and text editors for coding. The programming paradigms unit requires access to a range of programming languages, including Java, Python, C++, Javascript and Haskell. Jupyter Notebooks or Google CoLab could be used for collaborative work in coding
- coding practice platforms such as HackerRank or LeetCode could be helpful to gain programming skills
- for AI and mathematical computations, you should make available libraries such as PyTorch, Keras, Scikit-learn, NumPy, or SciPy. You should provide frameworks such as Node.js, React, Angular or Flask for the Web Development unit
- for the Database unit, you should provide tools such as MySQL, MongoDB, PostgreSQL or SQLite. For the Virtualization Technology unit, potential tools that you might use include VMware, Oracle VirtualBox, Hypervisor, and Docker
- for the Networking unit, you require network simulators such as Cisco Packet Tracer or GNS3
- to deliver the Applied Mathematics in Computing unit, you might supplement your teaching with mathematical software such as MATLAB or Mathematica
- for group projects and communication, you could make use of collaboration tools like Google Workspace or Microsoft Teams
- you should provide links as appropriate to websites, such as blogs, tutorials or online courses to support independent learning

Information for centres

Equality and inclusion

The units in this HND are designed to be as fair and as accessible as possible with no unnecessary barriers to learning or assessment.

You should consider the needs of individual learners when planning learning experiences, selecting assessment methods or considering alternative evidence.

Guidance on assessment arrangements for disabled learners and those with additional support needs is available on the assessment arrangements web page.

Internal and external verification

You must make sure all instruments of assessment you use in this qualification are internally verified according to your centre's policies and SQA's guidelines.

SQA carries out external verification to ensure that internal assessment meets the national guidelines for this qualification.

More information on internal and external verification is available in SQA's Guide to Assessment and in Next Generation: Higher National Quality Assurance — Guidance for Centres.

Glossary

SQA credits: 1 SQA credit equals 8 SCQF credit points.

SQA credit value indicates the contribution the unit makes to an SQA qualification. An SQA credit value of 1 represents approximately 40 hours of learning, teaching and assessment.

SCQF: the Scottish Credit and Qualifications Framework (SCQF) is Scotland's national framework for describing qualifications. We use SCQF terminology in this guide to refer to credits and levels. For more information on the SCQF, visit the SCQF website.

SCQF credit points indicate the amount of learning required to complete a qualification. NextGen HNCs and HNDs are worth 120 SCQF credit points.

SCQF levels indicate how hard the qualification is to achieve. The SCQF covers 12 levels of learning. NextGen HNCs are at SCQF level 7 and NextGen HNDs are at SCQF level 8.

Information for learners

HND Computer Science

This information explains:

- what the qualification is about
- what you should know or be able to do before you start
- what you need to do during the qualification
- opportunities for further learning and employment

Qualification information

The Higher National Diploma (HND) Computer Science provides you with the opportunity to gain the high-quality knowledge and skills you need to pursue a career in the ever-evolving field of modern computing and IT technologies. The demand for computer science professionals remains high as business and industry increase their requirement to sustain and develop applications of computing technologies. You understand that while there are many specialisms in the field of computing, there are fundamental concepts and technologies that underpin them. By completing this HND, you are well-equipped to pursue a computing specialism of your choice or undertake further study of computer science at degree level.

In HND Computer Science, you gain understanding of important aspects of computer science, including programming, data structures, algorithms, and computer systems. You can also specialise in a particular area of computer science, such as artificial intelligence (AI), cyber security or data science.

The specific aims of the qualification are that you can:

1. understand the key concepts and principles of computer science
2. appreciate current and emerging technologies in computer science, including AI and machine learning
3. develop mathematical and statistical understanding as it relates to computer science
4. apply knowledge of principles and concepts to real-world problems
5. prepare for employment in the design, development, testing and implementation of computing solutions
6. select and apply the correct approaches to resolving problems through computing solutions
7. develop computational thinking, pattern recognition, deconstruction, logical thinking, synthesis and analysis skills
8. develop professional practices and behaviours associated with computer science
9. gain understanding of the ethical, social and legal issues associated with the use of computer systems. including those relating to diversity, inclusion and sustainability
10. prepare for progression to further studies in computer science, or related disciplines, at SCQF level 9

During the course you carry out two mandatory units that add to 7 SQA credits. In the Computer Science unit, you learn the concepts and principles that are fundamental to the practice of computer science. In the Professional Practice unit, you work in a team to carry out an extended project, such as research into a topic in computer science or developing a computing solution to a problem from contexts such as science, engineering, health, business or cybersecurity.

Before you begin, you should have a good understanding of basic concepts of computer science, such as computer programming, computer hardware and software, along with good digital skills. You should also be familiar with software tools that aid the planning and reporting on your progress. You can evidence this by having an HNC qualification in computing, or equivalent at SCQF level 7.

To achieve the HND in Computer Science, you must pass the two mandatory units and achieve at least 8 SQA credits from a group of optional units. You can select optional units relating to a specialist area of computing.

You are assessed through a variety of approaches that reflect modern practice in assessment, including the use of projects, assignments, and investigations, as well as question papers. You are encouraged to use a range of media to present evidence, such as video, audio, web pages and social media platforms.

Your final award (provided all units are passed) is graded in a manner that reflects the quality of your work over the course. Project work that you have completed in the units of the course are graded according to a set of criteria that relate to the competences expected of a computer scientist.

Throughout the HND Computer Science you also develop your personal meta-skills while studying industry and sector-specific content. Meta-skills are higher-order skills that support the development of other skills and promote success in any context. They enable you to respond to professional challenges and opportunities by reflecting on, developing, applying and adapting industry skills and sector knowledge. These meta-skills are grouped into three categories: self-management, social intelligence, and innovation.

In line with government policy, you also develop your knowledge, skills and values related to the challenges of sustainability. You are encouraged to ask questions, analyse, think critically and work with others to make positive contributions to a sustainable future.

Successful completion of the HND Computer Science opens up career opportunities in a junior computing role. It also prepares you for progression to a degree-level qualification in computer science or further study in related computing topics.

Administrative information

Published: September 2025 (version 1.0)

History of changes

Version	Description of change	Date

Please check SQA's website to ensure you are using the most up-to-date version of this unit.

If a unit is revised:

- no new centres can be approved to offer the previous version of the unit
- centres should only enter learners for the previous version of the unit if they can complete it before its finish date

For more information on NextGen: HN Qualifications please visit the NextGen: HN web page.