Subject Benchmark Statement

Computing

March 2022

Contents

About this Statement

This document is a QAA Subject Benchmark Statement for Computing that defines what can be expected of a graduate in the subject, in terms of what they might know, do and understand at the end of their studies. Subject Benchmark Statements also describe the nature and characteristics of awards in a particular subject or area. Subject Benchmark Statements are published in QAA's capacity as a membership organisation on behalf of the higher education sector. A summary of the Statement is also available on the QAA website.

Key changes from the previous Subject Benchmark Statement include:

- a revised structure for the Statement which includes the introduction of cross-cutting themes of:
  - equality, diversity and inclusion
  - education for sustainable development
- employability, entrepreneurship and enterprise education
- a comprehensive review updating the context and purposes of Computing, including course design and content in order to inform and underpin the revised benchmark standards
- the new Statement covers bachelor's, integrated master's and postgraduate taught master's degrees in a single unifying Statement.

How can I use this document?

Subject Benchmark Statements are often used by higher education providers in the design and development of new courses in the relevant subject, as they provide a framework for specifying intended learning outcomes in an academic or vocational discipline. They are also used as a reference point when reviewing or revalidating degree courses. They may be used by external examiners in considering whether the design of a course and the threshold standards of achievement are comparable with other higher education providers. They also provide professional, statutory and regulatory bodies (PSRBs) with the academic standards expected of students.

Subject Benchmark Statements provide general guidance for articulating the learning outcomes associated with a course but are not intended to represent a national curriculum in a subject or to prescribe set approaches to teaching, learning or assessment. Instead, they allow for flexibility and innovation in course design within a framework agreed by the subject community.

You may want to read this document if you are:

- involved in the design, delivery and review of courses in Computing
- a prospective student thinking about undertaking a course in Computing
- an employer, to find out about the knowledge and skills generally expected of Computing graduates.

Relationship to legislation and regulation

The responsibility for academic standards lies with the higher education provider who awards the degree. Higher education providers are responsible for meeting the requirements of legislation and any other regulatory requirements placed upon them by their relevant funding and regulatory bodies. This Statement does not interpret legislation, nor does it incorporate statutory or regulatory requirements.

The regulatory status of the Statement will differ with regard to the educational jurisdictions of the UK. In England, Subject Benchmark Statements are not sector-recognised standards as set out under the Office for Students' regulatory framework. However, they are specified as a key reference point, as appropriate, for academic standards in Wales under Quality Assessment Framework for Wales and in Scotland as part of the Quality Enhancement Framework. Subject Benchmark Statements are part of the current quality requirements in Northern Ireland. Because the Statement describes outcomes and attributes expected at the threshold standard of achievement in a UK-wide context, many higher education providers will use them as an enhancement tool for course design and approval, and for subsequent monitoring and review, in addition to helping demonstrate the security of academic standards.

Additional sector reference points

Higher education providers are likely to consider other reference points in addition to this Statement in designing, delivering and reviewing courses. These may include requirements set out by PSRBs and industry or employer expectations. QAA has also published Advice and Guidance to support the Quality Code which will be helpful when using this Statement, for example, in course design, learning and teaching, external expertise and monitoring and evaluation.

Explanations of unfamiliar terms used in this Subject Benchmark Statement can be found in QAA's Glossary. Sources of information about other requirements and examples of guidance and good practice are signposted within the Statement where appropriate.

1 Context and purposes of a Computing degree

Context

1.1 This Subject Benchmark Statement refers to courses of study in Computing, including:

- bachelor's degrees with honours
- integrated undergraduate master's degrees
- postgraduate degrees that are generalist in nature
- postgraduate degrees that are specialist in nature
- courses designed to be studied in part-time mode at both undergraduate and postgraduate levels
- undergraduate and postgraduate degree apprenticeships.

1.2 This version represents a content review and a collation of previous Subject Benchmark Statements that had separate documents for bachelor's, integrated undergraduate master's and postgraduate taught master's courses of study. It has been produced by a group of subject specialists drawn from, and acting on behalf of, the subject community. The Statement makes reference to the Association of Computer Machinery's (ACM) and Institute of Electrical and Electronics Engineers' (IEEE) Computing Curricula 2020 as a source for guidance on detailed curriculum content.

1.3 It also provides a reference point for multidisciplinary courses that provide a joint programme with a substantial core of computing, for example, business and computing and data science.

Purposes of a Computing degree

1.4 The reasons for studying computing are as diverse as its domains of application. Some students are attracted by the depth and intellectual richness of the theory, others by the possibility of engineering large and complex systems. Many study computing for vocational reasons, or because it gives them the opportunity to explore creative and dynamic technologies. Many also study to improve their employment prospects in a rapidly evolving global digital skills economy. Whatever the perspective, computing can claim characteristics that, while present in other disciplines, are rarely present in such quantities and combinations. Particular emphasis has been placed within this Subject Benchmark Statement on meeting the needs of this rapidly evolving global digital skills economy through providing more equitable processes for learning, and subsequently earning, and, in so doing, addressing the United Nations Sustainable Development Goals, in particular 4, 5 and 8, which emphasise the need for inclusive lifelong learning, gender equality and sustainable economic growth, respectively.

Characteristics of a Computing degree

1.5 Computing is concerned with the understanding, design and exploitation of computation and computer technologies. It is a discipline that:

- blends elegant theories (including those derived from a range of other disciplines such as mathematics, engineering, psychology, graphical design or well-founded experimental insight) with the solution of immediate practical problems
- underpins the development of both small and large-scale systems that are secure, reliable, usable and that support organisational goals
- helps individuals in their everyday lives and realise their career aspirations

- is pervasive, ubiquitous and diversely applied to a range of applications, and important components are often invisible to the naked eye.

1.6 Computing promotes innovation and creativity. It requires a disciplined approach to problem solving. It approaches design and development through selection from alternative possibilities justified by carefully crafted arguments. It controls complexity first through abstraction and simplification, and then by the integration of components. Above all, it is a product of human ingenuity and provides major intellectual challenges, yet this limits neither the scope of computing nor the complexity of the application domains addressed.

1.7 Computing as a discipline is attractive to innovation, and this can equally arise from the foundational intellectual areas, for example algorithmics and cryptography, as from technology-driven opportunities.

1.8 It is hardly surprising that the diversity of computing is reflected in the varied titles and curricula that higher education providers have given to their computing-related degree courses. While this Statement aims to capture the nature of computing as a discipline, individual higher education providers may need to draw on a wider range of materials and resources, including ethics and data regulation or other Subject Benchmark Statements, to capture fully the specific character of their particular degree courses.

1.9 Computing degrees will continue to evolve in response to developments in the subject area and to reflect changes in the school curriculum. This Statement therefore concentrates on general graduate outcomes and does not specify a core computing curriculum.

1.10 Computing degrees often integrate a period of time working within a company or similar organisation, as an intern or placement student. Placements offer the opportunity for students to apply and validate their learning and skills in the context of the world of employment and to provide early exposure to the development of professional competence as enshrined within the skills expressed in the particular course of study.

Equality, diversity and inclusion

1.11 The pervasiveness of the computing discipline can enable a curriculum which is relevant and authentic, relates to real and current challenges, and able to promote greater social justice and equity. The curriculum should also speak to and be valued by every student, while addressing issues that are important to them. Culturally responsive computing approaches that recognise and value students' cultures can bring about ways for students to reflect and engage with issues of representation, exclusion, disadvantage and structurally embedded advantage. The curriculum should engage students in meaningful, culturally-contexed practical tasks in a welcoming and collaborative environment, as well as ensuring that technological solutions do not emerge with unintentional bias and limited insight into the diversity of the people who will develop and use them.

1.12 Curricula should therefore actively represent all students, acknowledging and removing existing biases, and enabling and supporting all students equally. For example, traditionally, the computer science field has suffered from gender imbalance, in part as a result of structurally genderised terminology, approaches to teaching and learning and the historical context in which the subject has developed.

1.13 Students can be significantly impacted by the way courses are structured, delivered and assessed. Understanding these potential impacts and how best to address them, through making reasonable adjustments, is key to providing a course that meets the needs of all students. For example, providing accessible content ahead of learning activities, supporting all students with both individual and group learning activities, ensuring personalised learning support is effectively combined with learning activities and providing different options for how learners can engage with the learning content all help to address such impacts.

1.14 The statements below provide an auditable list of equitable practices and processes by which greater equality and accessibility can be achieved. Course and module content and delivery may be mapped to these statements (Appendix 1), along with supporting examples and evidence, to demonstrate the processes by which greater equality, diversity and inclusion can be achieved.

Courses of study and their providers could consider:

- how to demonstrate due regard for unlawful and/or inappropriate conduct with consideration given to people with a protected characteristic (age, disability, gender reassignment, pregnancy and maternity, race, religion or belief, sex and sexual orientation)
- means for mitigating disadvantages suffered by people with a protected characteristic where these are different from the needs of other people, enable people within protected groups to take part in activities where their participation is disproportionately low and foster good relations between people who share a protected characteristic and those who do not
- equitable measures that discernibly address how teaching, learning and assessment promotes equality in relation to legislative protected characteristics, including special educational needs, age, disability (overt and hidden), gender reassignment, pregnancy and maternity, race, religion or belief, sex and sexual orientation as well as being sensitive to and promoting a curriculum that is broadly informed
- providing a variety of perspectives on computing topics and ensuring that case studies used as exemplars or as assessments are drawn from a diverse range, highlighting global perspectives, featuring diverse cultures and communities, showcasing the historical and cultural integration of computing and the diverse people who do computing
- providing inclusive opportunities for collaborative and team-based problem-solving where students can share, and possibly use as drivers for problem-solving, their own experiences and perspectives, without judgement or prejudice
- the development of the wider skills and behaviours for professional communication, collaborative working and reflective practice that are respectful, accessible and inclusive of all, at both the intercultural and intracultural; that is, between cultures and within a culture
- providing opportunities to involve students as co-creators of the curriculum to ensure that the vocabulary of computing addresses non-inclusive language that can be seen as confronting by under-represented groups
- providing inclusive and engaging teaching, learning, assessment and feedback strategies to support all students' development, success and employability, while connecting with their identities, experiences and cultural capital
- providing opportunities to democratise the physical and virtual learning spaces so that they are equally accessible and inclusive for all students
- the strategies required to mitigate or remove biases at each stage of the system life cycle, or other approaches adopted in the context of the curriculum being delivered, including developing reflective practice on own biases

- addressing the differential and social impact of technology in relation to people who share a protected characteristic and those who do not, paying attention to intersectionality and lack of representation
- acknowledging and addressing how divisions and hierarchies of colonial value are replicated and reinforced within the computing subject. For more information, please see Decolonising of curriculum
- highlighting methods of ethically aligned participatory design and human-centred computing that embed all stakeholders in the process
- demonstrating areas in software engineering and application development that require a professional and reflective approach to addressing bias and lack of diversity in data and design of computing solutions, systems and protocols.

Sustainability

1.15 Computing courses should address the sustainable development agenda through highlighting key economic, social and environmental issues that the discipline influences, whether positively or negatively. Courses should enable students to develop core personal and professional competencies to address the key societal challenges highlighted by the United Nations Sustainable Development Goals for 2030 in their future working lives. The Education for Sustainable Development Guidance (QAA and Advance HE, 2021) outlines pedagogic approaches for implementation in UK higher education institutions.

1.16 Across the Computing curriculum, we should be considering the broad definition of sustainable development, which encompasses economic and social sustainable development as well as environmental, and thinking about how the material that we teach links to these goals. There will be places where environmental sustainability can be discussed, for example, the resource consumption of massive data centres used for cloud computing, and more generally, the environmental costs of both building hardware to support computing and disposing of electronic waste. Sustainable development also links with equality, diversity and inclusion; therefore, issues around social justice and economic prosperity can also be brought into the discussion in many areas of computing, alongside developing curricula that are global in outlook and that support inclusive lifelong learning and economic prosperity for all.

1.17 There are 17 UN Sustainable Development Goals, but there are close relationships between them, and identifying one goal that is relevant is likely to lead to more. This Subject Benchmark Statement does not attempt to prescribe exactly which goals should be discussed in a Computing degree course, because computing is such a diverse field. The introduction of Education for Sustainable Development (ESD) is still emerging within the computing discipline, with providers exploring the effectiveness of alternative approaches, with some specific examples of practice being presented in Appendix 2. These are intended to be illustrative rather than prescriptive; consider how these ideas would apply to the particular focus of a specific Computing degree. In broad terms, commonly adopted approaches include:

- specific modules which focus directly upon Sustainability or closely related areas such as green IT
- embedding Sustainability as a key theme in appropriate areas such as professional issues related delivery
- addressing Sustainability in an integral manner as cross-curricula concern with relevant issues being flagged as their relevance emerges, such as within

requirements for engineering, cloud computing, machine learning, green IT as relates to infrastructure

- addressing Sustainability indirectly via the use of case studies/projects that address other aspects of the curricula but in doing so cover education for sustainable development.

1.18 Irrespective of the approach adopted, the intention is that Computing graduates should have an appreciation of domain-relevant issues related to sustainable development and the emergent manner by which these issues might be addressed.

Entrepreneurship and enterprise education

In this context, entrepreneurship and enterprise education are interpreted as referring to employability in its widest context.

1.19 Preparing computing students to move successfully from education into employment is an essential part of a degree course. While there are other departments within institutions that will support students in taking the next steps in their career together with a personal responsibility for the student themselves to take ownership, it is imperative that the development of employability skills is embedded into Computing courses, with the following section acting as a guide. Entrepreneurship and enterprise education are important factors to consider in relation to employability skills because their specific inclusion in degree curricula contributes to raising the employability levels of Computing graduates and the section below should be read in conjunction with Enterprise and Entrepreneurship Education: Guidance for UK Higher Education Providers (QAA, 2018).

Personal, professional and academic skills

Generic skills

- Ability to work with professional and behavioural integrity.
- Awareness of the legal and ethical factors and principles in the professional environment of computing.
- Awareness of sustainability issues related to the computing discipline/profession.
- Demonstrable understanding of computing as a global profession.
- Responsible use and application of social media in a professional environment. This to include professional online communication, email etiquette and the development of an online presence.
- Critical reflection skills and the ability to communicate effectively in writing, verbally and electronically.
- Apply a high standard of numeracy and literacy.
- Ability to interact with others in a professional and respectful way.
- Understand and demonstrate respect for the values of equality and diversity across a range of different industries and business sectors within the broad computing professional workplace (see Equality, diversity and inclusion on page 4).
- Understand the role of a leader in setting directions and taking responsibility for actions and decisions.
- Communicate with confidence at face-to-face and virtual meetings.

Agile working

- Ability to work with agility and flexibility in response to changing situations and priorities.

- Ability to interpret data and information within both academic and professional contexts in a timely way.
- Ability to recognise and make best use of the skills and knowledge of individuals to collaborate.
- Ability to identify problems and desired outcomes and negotiate with colleagues to achieve mutually acceptable solutions.

Remote working

- Ability to work unsupervised, plan effectively and meet deadlines.
- Ability to act confidently on one's own initiative.

Resilience

- Build social capital and apply professional networking skills.
- Maintain an up-to-date critical understanding of the developments, challenges and opportunities of the professional environment, both nationally and globally.
- The ability to succinctly present rational and reasoned arguments that address a given problem or opportunity, to a range of audiences (orally, electronically or in writing).

Entrepreneurship and enterprise education

- Active engagement in continuous professional development.
- Contribute to the development of the profession and ability to apply an enterprising and challenging approach.
- The ability to locate and retrieve relevant ideas and ensure that these can be effectively communicated in a variety of formats to a variety of audiences.
- Recognise factors in environmental and societal contexts which present opportunities and challenges for computing systems across a range of human activities.

Subject-specific skills in computing (common core)

- The subject-specific computing knowledge and skills specified elsewhere in this Subject Benchmark Statement will form the basis of the body of knowledge that an employer could reasonably expect a Computing graduate to be able to use and apply.
- Graduates should be able to link subject knowledge to business and industry by discussing examples and cases studies of how their subject knowledge of computing can be used by organisations.
- Application of computing knowledge and computational thinking to solve complex problems to the benefit and enhancement of the organisation or enterprise that the graduate will be working in.
- Ability to continuously plan and record self-learning and development in core computing skills and applications of technology as the foundation for lifelong learning and continuous professional development.

Developing employability skills through interdisciplinary thinking

- Computing is widely taught in joint and interdisciplinary courses, for example, management information systems, health informatics, computer science with economics, and scientific computing, for which it may be appropriate to draw on several Subject Benchmark Statements.

- Interdisciplinary teamwork and creative thinking are important employability skills for computing students. Interdisciplinary thinking promotes innovation, open-mindedness and creativity. Due to the interdisciplinary nature of computing, graduates should be able to apply computer science, data science and digital technology to problem-solving in other disciplines.
- Interdisciplinary project-based or problem-based learning in Computing courses could be used to encourage students to explore content beyond computing subjects, to draw insights from diverse disciplines and to apply them to the area of focus to solve problems by integrating knowledge and experience which come from computing and other disciplines.

Career prospects and employers' perspective

Career prospects

- There is a huge range of different employment pathways open to graduates of computing across organisation types: public and private sector, start-ups, scale-ups, small to large enterprises; and role discipline: software engineer, cybersecurity, data ethics; the list continues to grow. Graduates of computing should build an understanding of potential employment pathways, their differing skills requirements together with a self-awareness of personal skillsets, covering both technical and more generic employability skills and how to match these to career aspirations.
- Graduates should have the ability to communicate effectively how study and experience provides evidence of personal strengths and skills and how these can be effectively articulated to prospective employers, including the use of social media platforms to support professional profile development.
- Careers within computing have a high requirement for continuing professional development (CPD) and courses should encourage students to explore how to access CPD and how to build this into future career planning.
- Course planning and development should include working with employers wherever possible, to identify co-curricular activities such as bootcamps, hackathons, master classes and the development of authentic assessment opportunities.

Employers' perspective

- Courses should encourage students to explore the potential role of digital technologies within the workplace and how to apply appropriate digital technologies, techniques and principles to advance digital capabilities.
- Employers will expect graduates to be able to demonstrate core skills required to apply and advance digital technologies within the workplace and the development of these skills should be included within courses.
- Collaboration and teamwork: capability to work with other people, from a range of cultures, to achieve common goals, for example through group work, group projects and group presentations.
- Self-management: a readiness to accept responsibility and flexibility, to be resilient, self-starting and appropriately assertive, to plan, organise and manage time.
- Self-reflection: self-analysis and an awareness/sensitivity to diversity in terms of people and cultures, showing emotional intelligence and empathy.

- Communication and listening: including the ability to produce clear, structured communications using a variety of media.
- Capability advancement: including networking with peers and horizon scanning to identify opportunities to advance digital capabilities and create a better future.

2 Distinctive features of a Computing degree

Design

2.1 The term computing applies to an increasingly diverse set of degree courses, all based on the foundations of computer science. This Statement identifies computer science, computer engineering, software engineering, information technology, information systems, cybersecurity and data science as discipline areas and outlines the content covered by these (Appendix 3). A Computing degree may include subject matter from more than one discipline area.

2.2 This Statement does not specify a syllabus or include a body of knowledge. ACM, in conjunction with IEEE and other professional societies, maintain and regularly update curricula in several discipline areas (Appendix 3). These documents should be used to inform course design and curriculum content.

2.3 Many higher education providers deliver degrees focused on specific aspects of computing, for example computer networking, games, multimedia, artificial intelligence and health informatics. These courses count as computing if their content is informed by one or more of the discipline areas listed above. The mere fact that computers are deployed to solve problems in a certain area does not itself make that area fall within the field of computing.

2.4 Additionally, computing is widely taught in joint and interdisciplinary courses for which it may be appropriate to draw on a number of Subject Benchmark Statements. This Statement is the reference point for the computing component of such courses.

2.5 The title of a course cannot describe the whole of its content. However, course titles are not divorced from graduate knowledge, skills and abilities. There are natural overlaps between the different identified discipline areas and course specifications indicate careers the course's graduates would be expected to proceed into. For example, a degree in software engineering could include aspects of computer science, for example, formal methods; and information technology, for example, user advocacy. However, the title and course specification of the degree references the curriculum from the dominant discipline area.

2.6 The following are fundamental requirements associated with all degree courses in Computing:

- the topic and learning outcomes are identified and defined clearly, and their relationship to the subject of computing and its applications is carefully captured in the title of the award
- courses are carefully designed to accommodate students who enter with a wide range of accepted entrance qualifications
- the relevant theoretical underpinnings, which may or may not be mathematical in nature, are identified and should result in emphasis on those fundamental aspects of a subject which do not change in the context of rapid technological development
- the curriculum demonstrates an integration between theory and practice as well as the planned development of a set of attitudes and an appreciation of a range of applications and their impact on users
- there is an appropriate integration between a set of classes that demonstrates cohesion in content and a planned approach to the topic of the course

- a major component is a substantial individual activity, or team activity that provides substantial individual responsibility, that requires an awareness of material from across the individual modules and which provides opportunities for students to demonstrate a range of abilities and achievements
- all undergraduate and master's degree courses will meet the outcomes of the qualification descriptors identified in The Frameworks for Higher Education Qualifications of UK Degree-Awarding Bodies.

Accessibility

2.7 In any discussion of accessibility in computing education, it is important to recognise the unique responsibility that computing as a subject has to people with disabilities. Technology can be transformational: tasks that previously were difficult or impossible for some people can be made routine with the use of technology. However, technology can also cause barriers to access, and care should be taken to ensure technology acts to enable people rather than disable them.

2.8 As such, when we discuss accessibility in respect of computing education it must encompass both the teaching and practice of universal access, following standards and ensuring compatibility with assistive technologies. In terms of curricula, accessibility should be treated as a standard part of computing. For example, where software development lifecycles are taught, accessibility should be embedded as a concern along with other concerns, such as the testing of software quality.

2.9 In terms of practice, curricula should be designed with accessibility in mind, meeting statutory requirements as a minimum. Creating educational environments with accessibility as a primary concern can enable students to study computing when without such consideration, they would be unable to do so. For example, if a course requires students to regularly attend a physical location on campus, this may place barriers in the way of some students.

Progression

2.10 Over the course of a degree with honours (FHEQ Level 6; FQHEIS Level 10) a Computing student will progress from one year of study to the next, in line with the regulations and processes for each institution. However, it is expected that each year would see the attainment of certain levels of knowledge, expertise and experience which builds towards the final achievement of meeting all of the threshold-level subject-specific and generic skills listed in this Statement. Upon graduation from an undergraduate degree, it would be expected that a student who had achieved a second-class degree or higher would be capable of, and equipped for, undertaking postgraduate study in computing or an associated discipline.

2.11 Joint honours undergraduates will achieve elements of the specific and generic skills for the subject but will add others according to the subjects covered in a joint programme.

2.12 Integrated undergraduate master's degrees (FHEQ Level 7; FQHEIS Level 11) are available in the UK and typically comprise a four-year full-time course or a part-time course of not less than five and not more than eight academic years. Students exiting earlier may be eligible for a Certificate of Higher Education, a Diploma of Higher Education or an honours degree depending upon the years of study completed to a satisfactory standard. Similarly, in a standard three-year undergraduate honours degree qualification, students may also exit earlier with a Certificate or Diploma depending upon their achievements. Scottish bachelor's degrees with honours are typically designed to include four years of study, which relates to the structure of Scottish primary and secondary education.

Flexibility

2.13 Flexible educational approaches enable learners to adapt their education to their situational and contextual individual needs and constraints. Flexible educational approaches may also play a key role in increasing access into further and higher education and social mobility. For example, such approaches could provide scope for learners to select educational opportunities that are better suited to their current level of proficiency and interests.

2.14 To this end, there may need to be:

- flexible delivery modes, including but not limited to campus-based, online distance learning, block release and hybrid (campus-based and online)
- flexible study patterns in terms of intensity of study and start dates
- flexible approaches to assessment tasks that enable learners to demonstrate different competencies
- greater focus on assessment that is authentic to learners; this may include project-based tasks and teamwork, while maintaining academic rigour and theoretical foundations
- more flexible approaches to credits, for example the integration of micro-credentials with traditional modular credits
- flexible approaches to recruitment processes that recognise prior learning (accreditation of prior certificated learning, APCL) and/or work experience in the computing field (accreditation of prior experiential learning, APEL)
- designing courses as online first may be useful in reducing and avoiding barriers to flexible approaches
- sufficiently flexible processes to develop and review courses of study so that course teams are able to dynamically address the needs of learners, industry and society.

Partnership

2.15 Many degree courses provide opportunities for work-based learning such as apprenticeships, placements and live work-based projects. Other courses may offer opportunities for study abroad. These opportunities are usually achieved through the higher education provider working in partnership with other external organisations.

2.16 Further advice and guidance can be found on Partnerships and Work-based Learning.

2.17 Computing courses can benefit from other collaborations with external organisations in activities such as curriculum design, talks, workshops, projects and visits. An effective external/industry advisory board can help ensure courses are relevant to the needs of the computing sector and advise on the development of employability skills.

Monitoring and review

2.18 A major feature of academic quality assurance and enhancement at a higher education institution is having in place monitoring and regular review processes for the courses it delivers. Degree-awarding bodies routinely collect and analyse information and undertake periodic course review according to their own needs. They will draw on a range of external reference points, including this Statement, to ensure that their provision aligns with sector norms. Monitoring and evaluation is a periodic assessment of a course, conducted internally or by external independent evaluators. Evaluation uses information from current and historic monitoring to develop an understanding of student achievement and inform future course planning. A combination of continuous and retrospective monitoring practices is vital for the effective monitoring and review of courses of study.

2.19 Externality is an essential component of the quality assurance system in the UK. Higher education providers will use external reviewers as part of periodic review to gain an external perspective on any proposed changes and ensure threshold standards are achieved and content is appropriate for the subject.

2.20 External examination currently in use across the UK higher education sector also helps to ensure consistency in the way academic standards are secured by degree-awarding bodies. Typically, external examiners will be asked to comment on the types, principles and purposes of assessments being offered to students. They will consider the types of modules on offer to students, the outcomes of a cohort and how these compare to similar provision offered within the UK. External examiners produce a report each year and make recommendations for changes to modules and assessments (where appropriate). Subject Benchmark Statements, such as this one, can play an important role in supporting external examiners in advising on whether threshold standards are being met in a specific subject area.

2.21 Courses with professional and vocational outcomes may also require evaluation and accreditation from professional and regulatory bodies. These are usually done through a combination of site visits and desk-based reviews.

2.22 Additionally, courses should also consider the inclusion of learners in their monitoring and review processes as part of their student-staff partnership work.

3 Content, structure and delivery

Content

3.1 Computing is any purposeful activity using, exploiting or creating computing equipment and services. As equipment and services rapidly change there is an increasing need to shift from experience and end qualification-based recruitment processes to more fluid, talent-based approaches where badges, micro-credentials, skills-based hiring and lifelong learning play an increasingly important role in enabling global learner-earners to move more seamlessly between education and employment.

3.2 Within this context, undergraduate and postgraduate degrees will continue to represent important milestones for learners entering the workforce, but they are also expected to play a more important role in reskilling the workforce as economic requirements change. As more granular micro-credentialing plays an increasing role in meeting employer needs there is also an increasing emphasis on both undergraduate and postgraduate degrees continuing to meet the needs of employers and learners as springboards into careers, both initially and as learner-earners adapt and retrain both in the UK and overseas. Providing content that can be added to learner-earners' existing achievements, without duplication and unnecessary additional study, will therefore become increasingly important in supporting an adaptive global workforce.

3.3 Traditionally, computing stakeholders have included employers, professional bodies (representing employer expectations as well as providing aligned training), cross-sector educational institutions and regulatory organisations. As education and employment becomes increasingly global, international organisations such as the OECD, UNESCO, World Bank and International Labour Organization also play an increasingly important role, especially in defining future educational and workforce requirements. Within this context it is important, in particular, to consider future skills requirements in the next 10 to 15 years, as discussed, for example, in the OECD Future of Education and Skills 2030 project, and the increasingly important role of twenty-first century skills within employment. Both point to a need to more clearly and effectively communicate what would traditionally be seen as non-technical capabilities and competencies alongside technical capabilities and competencies and to provide a global and skills-based perspective when contextualising the curriculum.

3.4 Pedagogically, educators should consider the balance between subject-specific and transferable skills development in both curriculum and assessment, opportunities for personalised learning and evidencing of individual capability and competency development within courses and the balance between educational and workforce requirements as courses are developed and maintained. The value of learning, in providing both learner-earner agency and value to society and the economy, should be both clear and wherever possible differentiated between learners. Problem-solving and artefact creation form core parts of the computing discipline and enabling learners to evidence these within a curriculum in ways they understand and can communicate with others are encouraged. The use of capstone activity as well as teamwork is encouraged within courses.

Teaching and learning

3.5 There is an increasing emphasis on being able to recognise, record and communicate capability and competency development and performance both within education and employment. In the context of undergraduate and postgraduate degrees, this highlights the increasing need, in particular, to evidence learning of computing competencies; that is, the application of computing capabilities in applied contexts. Typically, these contexts would be work-based, such as those traditionally provided through apprenticeships, internships, placements and live projects with clients. However, as the demarcation between education and employment becomes more blurred it is likely that other forms of competency measure may play an increasing role in recognition, such as online evaluations, role-playing scenarios and gig-economy/commissioned work.

3.6 Alongside a shift to more flexible recording and recognition of education and employment, there is also a need for individuals to become better able to control their own learner-earner journeys. Having agency over their personal performance requires both a better understanding of themselves and appropriate tools and techniques to enable them to self-regulate and to optimise their personal performance. As such, there is a greater emphasis on educational institutions to provide opportunities for reflection, performance monitoring, evaluation and feedback within learning to support a more personalised learning journey. Portfolios, accommodation of non-formal and informal learning, flexible, adaptive and personalised assessment approaches and better use of evaluative data such as careers information and learning analytics provide opportunities to support a more fluid and responsive learning experience.

3.7 Computing courses deploy a diverse range of teaching approaches, which is a strength of the subject. The teaching methods employed should have the potential to firstly deliver a range of learning that can be demonstrated through the achievement of learning outcomes, indicating capability. They should also ensure that learners have competence in specific areas of computing, which requires an ability to work independently in these areas.

3.8 Competence is often achieved through repeated exposure to practical coursework, both individual and group-work. While small projects can be developmental, competence typically requires work to be completed that is of a significant scale and/or complexity and that is undertaken in a real or simulated work context or is addressing real world requirements.

3.9 Teaching should encourage students to reflect, evaluate, select, justify, communicate and be innovative in their problem-solving; and prepare them to become adaptable independent learners throughout their lifelong learner-earner journey. Students should be exposed to a range of tools and techniques relevant to their course, including bespoke, open source and commercially available approaches. The diverse prior learning, capabilities, competencies and experiences should be understood and accommodated. Learning barriers should be minimised wherever possible to enable all learners to be as successful as possible. Equality, diversity and inclusion within the curriculum should be carefully considered as outlined in section 1.

3.10 Curriculum design should reflect appropriate research, scholarship and the ability to make an informed choice as to which is the most appropriate approach to use in a specific context, alongside current industrial and business practices. Teaching needs to be placed within the context of the legal, social, ethical, professional, environmental and economic factors that are relevant to computing. In addition to technical subjects such as security, the curriculum should use an inclusive design approach and also include relevant aspects of sustainability as well as equality, diversity and inclusion. The attitudes and values of graduates should also be developed to enable them to thrive in a diverse and rapidly changing subject such as computing.

3.11 To ensure students have sufficient underpinning, the early stages of undergraduate courses will typically expose students to a broad range of computing topics, becoming more focused in the final year. Generalist master's, where no prior subject knowledge may have been studied, will typically focus on meeting the minimum subject knowledge required of a Computing graduate. Generalist master's may be designed to offer a broad underpinning or to focus on particular niche areas such as software testing. Specialist master's (and the final year of integrated undergraduate master's) will typically be more focused on specific aspects of the subject. In addition to increasing focus, progression through Levels should also reflect a move from general underpinning to current research and state of the art. Further, it is important that progression reflects increasing levels of independent learning, so students learn how to learn for themselves, supporting their future development and enabling them to retain currency throughout their career.

3.12 The curriculum will define the knowledge students will gain and the course learning outcomes indicate the areas in which graduates will have knowledge competence or capability. However, individual students are expected to have the opportunity to develop a greater level of competence in some aspects of computing. These areas could also be linked to specific applied learning contexts, the particular focus of their major capstone activity, or a student's chosen pathway through optional modules on the course. Individual students could highlight their competencies, for example, in a learning portfolio, through the achievement of badges linked to micro-credentials, or in their curriculum vitae.

3.13 The hardware and software resources available should facilitate a practical approach to the delivery of the course. Staff delivering Computing courses should be able to demonstrate ongoing personal development in the subject and appropriate pedagogy. Such activity should be supported at institutional level. Normally, staff would be qualified to a level beyond the level they are teaching and would also have undertaken, or be undertaking, a teaching qualification relevant to higher education.

3.14 Institutions offering integrated undergraduate specialist master's courses must be able to articulate the legitimacy of providing education in the course topic. Although there are other possibilities, this will often mean that the institution possesses staff who are at the forefront of developments in the topic of the course and engaged in related advanced scholarship.

3.15 Computing graduates need to have acquired a strong grounding in underpinning knowledge alongside an ability to apply that knowledge to support future lifelong learning and the development of competence in new technologies as they are developed. Typically, students would demonstrate their ability to learn and apply a new approach or as part of the major activity that takes place towards the end of their course.

3.16 Course delivery should incorporate opportunities for applied learning in an authentic or simulated work context, such as industrial placements or apprenticeships. Working in teams of several people on sizable projects can simulate the real-world environment as well as exposing students to complexity. Projects defined in collaboration with industrial partners or research groups can enhance student learning and stimulate and promote interest in self-regulated learning as well as exposing students to aspects such as legal or ethical issues, demonstrating their relevance and importance.

Assessment

3.17 Assessment for Computing courses should be varied, appropriate and engaging. Perhaps most importantly, assessments should be authentic and tied to real-world contexts and constraints, and also allow students to practically demonstrate the skills they have developed.

3.18 Academic integrity is a key concept here and assessment design plays a critical role in enabling all students to succeed to the best of their abilities. Assessment techniques should afford students the opportunities to showcase what they can do, not just what they can write about, while also ensuring that checkpoints are in place to ensure that only work produced by students themselves is being assessed, for example by following the principles in the QAA's contract cheating guidance.

3.19 Computing courses often conclude with a capstone activity, which brings together knowledge and practical and analytical skills that learners have developed throughout the course. This may take the form of a traditional project or end-point assessment, but other formats can be appropriate, whether research or practice-led.

3.20 A traditional written dissertation may not be suitable for all Computing courses, so a capstone activity may also include the production of academic papers, the development of a professional portfolio or work on entrepreneurial activities. Such activities at master's level should have a greater focus on drawing upon research literature.

4 Benchmark standards

Introduction

4.1 This Subject Benchmark Statement sets out the minimum threshold standards that a student will have demonstrated when they are awarded an honours degree in Computing. Demonstrating these standards over time will show that a student has achieved the range of knowledge, understanding and skills expected of graduates in computing. Benchmark standards are defined for bachelor's degrees with honours, integrated undergraduate master's degrees, postgraduate master's degrees and degree apprenticeships in Computing.

4.2 The benchmark standards are defined relative to the appropriate FHEQ Level 6 or 7 (FQHEIS Level 10 or 11) specification and associated descriptors. As such, their application to an individual course is necessarily contextual. As Computing degrees are often accredited by UK professional bodies, for example through the British Computer Society, the Institute for Apprenticeships and Technical Education, sector skills bodies or other Engineering Council recognised accreditors, all professional competency standards will apply in many instances.

4.3 The vast majority of students will perform significantly better than the minimum threshold standards. Each higher education provider has its own method of determining what appropriate evidence of this achievement will be and should refer to Annex D: Outcome classification descriptions for FHEQ Level 6 and FQHEIS Level 10 degrees. This Annex sets out common descriptions of the four main degree outcome classifications for bachelor's degrees with honours: 1st, 2:1, 2:2 and 3rd. While it is expected these benchmarks will provide the foundation for awards, it is understood that variances may arise for reasons of institutional autonomy or subject evolution, and that any deviation from these standards be identified and justified.

Undergraduate standards: threshold, typical and excellent descriptors

4.4 Three levels of attainment are defined across a range of course outcome categories.

4.5 With regard to undergraduate courses, those graduating with a bachelor's honours degree in Computing must demonstrate at least a threshold level of attainment across all outcome categories. Threshold level attainment typically maps onto that associated with a 3rd class honours. The typical level descriptor defines outcomes usually associated with 2:1. or 2:2. class honours performance. The excellent descriptor defines outcomes usually associated with a 1st class honours. It is not expected that a 1st class honours student must demonstrate excellent attainment in all outcome categories.

4.6 See paragraphs 4.7-4.17 for details on how the threshold, typical and excellent attainment descriptors apply to integrated undergraduate master's degrees and postgraduate master's degrees.

	Threshold	Typical	Excellent
Subject knowledge, understanding and skills	Demonstrate a requisite understanding of the main body of knowledge for their subject	Demonstrate a sound understanding of the main body of knowledge for their subject and be able to exercise critical judgement in the use of that knowledge	Demonstrate an exceptional understanding of the main body of knowledge for their subject and be able to exercise insightful and critical judgement in the use of that knowledge. Be creative and innovative in the application of the principles covered in the curriculum, and be able to go beyond what has been taught in classes
Intellectual skills	Understand and apply essential concepts, principles and practices of the subject in the context of well-defined scenarios, showing judgement in the selection and application of tools and techniques	Critically analyse and apply concepts, principles and practices of the subject in the context of loosely defined scenarios, showing effective judgement and adaptability in the selection and use of tools and techniques	Critically analyse and apply a wide range of concepts, principles and practices of the subject in the context of open scenarios, showing refined judgement and adaptability in the selection and use of tools and techniques
Computational problem-solving	Be able to demonstrate judgement, critical thinking and problem-solving skills to solve well-specified problems, to create computational artefacts with a degree of independence	Be able to demonstrate detailed judgement, critical thinking and problem-solving skills to solve both well-specified and loosely defined problems, to create appropriate computational artefacts	Be able to demonstrate sophisticated judgement, critical thinking, research design, and well-developed problem-solving skills with a high degree of autonomy, and to create highly effective computational artefacts across complex and unpredictable circumstances
Practical skills across the computing lifecycle	Demonstrate the ability to undertake problem identification and analysis to appropriately design, develop, test, integrate or	Demonstrate the ability to undertake problem identification and analysis to appropriately design, develop, test, integrate or deploy a complex computing	Demonstrate the ability to undertake problem identification and analysis to appropriately design, develop, test, integrate or deploy a highly complex

	Threshold	Typical	Excellent
	deploy a computing system and any associated artefacts; understand the relationship between stages	system and any associated artefacts; understand the relationship between stages and be able to demonstrate related problem-solving and evidence-informed evaluative skills	computing system and any associated artefacts; deeply understand the relationship between stages and be able to demonstrate related sophisticated problem-solving and evidence-informed evaluative skills
Interpersonal and team working skills (see also Entrepreneurship and enterprise education)	Demonstrate the ability to work in an effective manner, including as a member of a team, making use of tools and techniques to appropriately communicate, manage tasks and plan projects under guidance	Demonstrate the ability to work in a proactive and effective manner, including as a member of a team, making good use of tools and techniques to successfully communicate, manage tasks and plan projects with minimum guidance	Demonstrate the ability to work in a highly proactive and accomplished manner, including as a leading member of a team, making excellent use of tools and techniques to proficiently communicate, manage tasks and plan projects with minimum guidance
Professional practice (see also Equality, diversity and inclusion, Sustainability and Entrepreneurship and enterprise education)	Identify appropriate practices and perform work within a professional, legal and ethical framework – including data management and use, security, equality, diversity and inclusion (EDI) and sustainability – in the work that they undertake	Identify appropriate practices and effect principled solutions within a professional, legal and ethical framework to address core considerations – including data management and use, security, equality, diversity and inclusion (EDI) and sustainability – in the work that they undertake	Identify best-of-kind practices and effect highly principled solutions within a professional, legal and ethical framework to consistently address a wide breadth of relevant considerations – including data management and use, security, equality, diversity and inclusion (EDI) and sustainability – in the work that they undertake

Postgraduate degrees

4.7 A broad range of postgraduate Computing degrees can be defined. With reference to the QAA characteristics statement on master's degrees, this includes specialised postgraduate degrees that provide learners with an opportunity to deepen their study, typically within a specific area. It also includes professional and practice-based postgraduate degrees, including those that may attract entrants from a diverse range of undergraduate qualifications.

4.8 Nonetheless, all master's degree graduates have in-depth and advanced knowledge and understanding of their subject and/or profession, informed by current practice, scholarship and research. This will include a critical awareness of current issues and developments in the subject and/or profession; critical skills; knowledge of professional responsibility, integrity and ethics; and the ability to reflect on their own progress as a learner.

4.9 Graduates of research master's are likely to be further characterised by their ability to study independently in the subject, and to use a range of techniques and research methods applicable to advanced scholarship in the subject. Graduates of specialist or advanced study master's are likely to be characterised in particular by their ability to complete a research project in the subject, which in some subjects includes a critical review of existing literature or other scholarly outputs. Meanwhile, graduates of professional or practice master's are able to apply research and critical perspectives to professional situations, both practical and theoretical, and to use a range of techniques and research methods applicable to their professional activities.

4.10 Graduates of all types of master's degrees are equipped to enter a variety of types of employment (either subject-specific or generalist) or to continue academic study at a higher level, for example a doctorate (provided that they meet the necessary entry requirements). Graduates of professional/practice master's courses in particular possess the skills and experience necessary for some professions or areas of practice.

4.11 The broad range of postgraduate Computing degrees entails that the full range of course outcome categories, as identified in Undergraduate standards: threshold, typical and excellent descriptors, may not apply to all courses.

4.12 In all cases, the content of the course should align to the FHEQ Level 7/SCQF Level 11 specification.

4.13 With reference to the descriptors outlined in Undergraduate standards: threshold, typical and excellent descriptors, students graduating with a postgraduate degree in Computing must demonstrate at least a threshold level of attainment across all relevant course outcome categories. Attainment at a threshold level usually maps onto that associated with a Pass award.

4.14 The typical level descriptor defines outcomes usually associated with the typical threshold performance across all relevant course outcome categories. Attainment at a typical level usually maps onto that associated with a Pass/Merit or Commendation award. The excellent descriptor defines outcomes usually associated with the award of a Distinction. It is not expected that such a student must demonstrate excellent attainment in all outcome categories.

	Typical	**Excellent**
Subject knowledge, understanding and skills	Demonstrate a systematic understanding of knowledge, as appropriate to the area of study, much of which is at, or informed by, the forefront of their academic discipline, field of study or area of professional practice. Alongside this, demonstrate a critical awareness of current problems and/or recent development within the discipline. Demonstrate a degree of originality in the application of knowledge, together with a practical understanding of how established investigative techniques of research and enquiry are used to create and interpret knowledge in the discipline.	Demonstrate a deep and systematic understanding of knowledge, as appropriate to the area of study, much of which is at, or informed by, the forefront of their academic discipline, field of study or area of professional practice. Alongside this, demonstrate a critical awareness and advanced understanding of current problems and/or recent development within the discipline. Demonstrate originality in the application of knowledge, together with a practical and reflective understanding of how established and state-of-the-art investigative techniques of research and enquiry are used to create and interpret knowledge in the discipline.
Intellectual skills	Be able to analyse, apply and critically evaluate concepts, principles and practices at the forefront of the area of study.	Be able to deeply analyse, apply and critically evaluate concepts, principles and practices at the forefront of the area of study, demonstrating insight and innovation, and application of these skills as appropriate.
Computational problem-solving	Be able to demonstrate judgement, critical thinking, research design, and well-developed problem-solving skills with a high degree of autonomy, and to create effective computational artefacts given complex or open constraints.	Be able to demonstrate innovation and/or originality, sophisticated judgement, critical thinking, research design and well-developed problem-solving skills with a high degree of autonomy, and to create comprehensive and highly effective computational artefacts given complex or open constraints.
Practical computing skills	Demonstrate the ability to apply computing techniques, as appropriate to the area of study, within complex or unpredictable scenarios, in a systematic manner, making appropriate decisions given incomplete or missing data.	Demonstrate the ability to apply state-of-the-art computing techniques, as appropriate to the area of study, within highly complex or unpredictable scenarios, in a systematic and creative manner, making insightful decisions given incomplete or missing data.
Self-directed learning and attainment	Demonstrate elements of self-direction in tackling and solving problems, alongside approaching	Demonstrate self-direction in tackling and solving complex problems, alongside approaching

	Typical	Excellent
	and implementing tasks and activities in a proactive and effective manner.	and implementing tasks and activities in a highly proactive and effective manner.
Professional practice	Ability to communicate their work to specialist and a diverse range of non-specialist audiences. Identify appropriate practices in complex and unpredictable professional environments, and perform work within a professional, legal and ethical framework – including data management and use, security, equality, diversity and inclusion (EDI) and sustainability – in the work that they undertake.	Ability to communicate their work to specialist and non-specialist audiences in an accessible and impactful way. Identify and effect best-of-kind and highly principled solutions in complex and unpredictable professional environments, within a professional, legal and ethical framework to consistently address a wide breadth of relevant considerations – including data management and use, security, equality, diversity and inclusion (EDI) and sustainability – in the work they undertake.

Integrated undergraduate master's degrees

4.15 Integrated undergraduate master's degrees (MComp, MEng and MSci) include and build upon the outcomes of bachelor's degrees with honours to provide a greater range and depth of outcomes. An integrated course of study typically involves study equivalent to at least four full-time academic years in England, Wales and Northern Ireland and five in Scotland (www.qaa.ac.uk/docs/qaa/quality-code/qualifications-frameworks.pdf, paragraph 4.17.4).

4.16 The descriptive characteristics associated with an integrated undergraduate master's degree as defined within the QAA Characteristics Statement on master's degrees are considered to apply within this Statement.

4.17 Students graduating with an integrated undergraduate master's degrees in computing must demonstrate at least a threshold level of attainment across all descriptors in both FHEQ Levels 6 and 7, or FQHEIS Level 10 and 11 in paragraphs 4.2 and 4.3.

5 List of references and further resources

QAA, 2021, Quality Enhancement Review
www.qaa.ac.uk/reviewing-higher-education/types-of-review/quality-enhancement-review

QAA, Quality Enhancement Framework
www.qaa.ac.uk/scotland/quality-enhancement-framework

Computing Curricular 2020
www.acm.org/binaries/content/assets/education/curricula-recommendations/cc2020.pdf

Decolonising the Curriculum
https://edta.info.yorku.ca/decolonizing-the-curriculum

QAA and Advance HE, 2021, Education for Sustainable Development
www.qaa.ac.uk/quality-code/education-for-sustainable-development

QAA, 2018, Enterprise and Entrepreneurship Education: Guidance for UK Higher Education Providers
www.qaa.ac.uk/quality-code/enterprise-and-entrepreneurship-education

QAA, 2014, The Frameworks for Higher Education Qualifications of UK Degree-Awarding Bodies
www.qaa.ac.uk/docs/qaa/quality-code/qualifications-frameworks.pdf

QAA, 2019, Characteristics Statement: Higher Education in Apprenticeships
www.qaa.ac.uk/docs/qaa/quality-code/characteristics-statement-apprenticeships.pdf

OECD Future of Education and Skills 2030 project
www.oecd.org/education/2030-project

QAA, 2020, Contracting to Cheat in Higher Education: How to Address Essay Mills and Contract Cheating
www.qaa.ac.uk/docs/qaa/guidance/contracting-to-cheat-in-higher-education-2nd-edition.pdf

The British Computer Society
www.bcs.org

Annex D: Outcome classification descriptions for FHEQ Level 6 and FQHEIS Level 10 degrees
www.qaa.ac.uk/quality-code/qualifications-frameworks

QAA, 2020, Master's Degree Characteristics Statement
www.qaa.ac.uk/docs/qaa/quality-code/master's-degree-characteristics-statement.pdf

6 Membership of the Advisory Groups for the Subject Benchmark Statement for Computing

Membership of the Advisory Group for the Subject Benchmark Statement for Computing and Computing (Master's) (2022)

Dr Alan Hayes (Chair)	University of Bath
Professor Nik Bessis	Edge Hill University
Georgia Clarke	QAA Coordinator
Sarah Clowe	Keele University
Professor Tom Crick MBE	Swansea University
Dr Allison Gardner	Keele University
Professor Paul Hanna	Ulster University
Professor Philip Hanna	Queen's University Belfast
Professor Alastair Irons	British Computer Society
Dr Wendy Ivins	Cardiff University
Kevin Kendall	QAA Officer
Dr Sandi Kirkham	Staffordshire University
Dr Thomas Lancaster	Imperial College of Science, Technology and Medicine
Dr Mariana Lilley	University of Hertfordshire
Tom Lovell	TechSkills
Professor Minhua Eunice Ma	Falmouth University
Dr Emma Norling	University of Sheffield
Dr Preeti Patel	London Metropolitan University
Professor Shushma Patel	De Montfort University
Professor Karen Petrie	University of Dundee
Professor Mark Perry	Brunel University
Dr Tom Prickett	University of Northumbria at Newcastle
Professor Rupert Ward	University of Huddersfield
Dr Ian Wilson	University of South Wales

QAA would like to thank Professor Elizabeth Cleaver, Professor Michael McLinden and the Disabled Students' Commission for their valued contributions to the development of the Statement.

Membership of the Advisory Group for the Subject Benchmark Statement for Computing and Computing (Master's) (2019)

The fourth edition, published in 2019, was revised by QAA to align the content with the revised UK Quality Code for Higher Education, published in 2018. Proposed revisions were checked and verified by one of the Co-Chairs of the Subject Benchmark Statement for Computing from 2016.

Dr Alan Hayes	University of Bath
Dr Alison Felce	QAA

Membership of the Advisory Group for the Subject Benchmark Statement for Computing (2015)

Details below appear as published in the third edition of Subject Benchmark Statement for Computing (2015)

Dr Phil Brooke	Teesside University
Professor Christopher Clare	BCS Academic Accreditation Committee

Dr Tom Crick	BCS Academy
Professor Sally Fincher (Co-Chair)	University of Kent
Alan Hayes (Co-Chair)	University of Bath
Dr Iain Phillips	Loughborough University
Dr Alan Tully	University of Newcastle
Corresponding members	
Professor Quintin Cutts	University of Glasgow
Professor Andrew McGettrick	University of Strathclyde
Robert Koger (Employer representative)	Vision Semantics and IET Academic Accreditation Committee
Emilia Todorova (Student reader)	Glasgow Caledonian University
Janet Bohrer	QAA
Dr Tim Burton	QAA

Membership of the original Advisory Group for the Subject Benchmark Statement for Computing (Master's) (2011)

Liz Bacon	University of Greenwich
Roger Boyle	University of Leeds
Laurence Brooks	Brunel University
Anne de Roeck	The Open University
Jeff Magee	Imperial College
Gerry McAllister	University of Ulster
Andrew McGettrick (Chair)	University of Strathclyde
Keith Mander	University of Kent
Elizabeth Friend (Coordinator)	BCS, The Chartered Institute for IT
Laura Bellingham (QAA Officer)	QAA

Membership of the Advisory Group for the Subject Benchmark Statement for Computing (2007)

Details below appear as published in the second edition of Subject Benchmark Statement for Computing (2007).

Dr Laurence Brooks	Brunel University
Graham Gough	The University of Manchester
Alastair Irons	University of Northumbria
Dr Gerry McAllister	University of Ulster
Professor Andrew McGettrick (Chair)	University of Strathclyde
Professor Keith Mander	University of Kent

Membership of the Advisory Group for the Subject Benchmark Statement for Computing (2000)

Details below appear as published in the original Subject Benchmark Statement for Computing (2000).

Professor J Arnott	University of Dundee
Professor D Budgen	Keele University
Dr PC Capon	University of Manchester
Mr G Davies	The Open University
Professor PJ Hodson	University of Glamorgan
Professor E Hull	University of Ulster
Professor G Lovegrove	Staffordshire University
Professor KC Mander	University of Kent at Canterbury

Professor A McGettrick (Chair)	University of Strathclyde
Mr P McGrath	Leeds Metropolitan University
Dr A Norman	University of Cambridge
Mr SJ Oldfield	University of Plymouth
Ms A Rapley	Kingston University
Professor VJ Rayward-Smith	University of East Anglia

Appendix 1 - Mapping and audit templates

1: QAA Computing Benchmark Statements mapping

This provides a mapping of the Subject Benchmark Statement for Computing to the EDI statements. The intent is not to cover all possible actions and solutions but to provide a prompt to how the statements can be meaningfully applied.

Subject Benchmark Statement	EDI statement	Explanation	Example

2: EDI audit template

The below EDI audit template allows schools and departments to map their strategies and course content to the EDI statements. It is advised that the template be adapted for school-level, programme-level and module-level mapping.

It is assumed that evidence of application of each EDI statement and sub-statements would occur multiple times throughout.

Strategy and content	EDI statement	Explanation	Evidence

Appendix 2 - Examples of ESD approaches

Illustrative examples of approaches to Education for Sustainable Development in current computing curricula.

Approach	Specialism	Further information
Specific module	Computer science	As an introductory or higher-level module, Ken Abernethy and Kevin Treu (2014) Integrating sustainability across the computer science curriculum, *Journal of Computing Sciences in Colleges*, vol 30, issue 2, pp 220–228
	Software engineering	B. Penzenstadler et al, *Everything is INTERRELATED: Teaching Software Engineering for Sustainability* (2018) IEEE/ACM 40th International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET), pp 153-162
	Computer science	Within module entitled 'The Environmental Impact of Modern Computing' in which students study the effect that computing has on the environment via mining and mineral extraction, manufacturing and transport, cloud computing and energy usage.
Key theme	Professional issues	Students consider how technology is part of the problem and part of the solution as part of a professional issues module. Taught material in a module on Professional Issues on the sustainability issues involved in the decisions we make when designing computing systems, and on how computing can be used to improve what we do and our understanding of what is happening. Gordon N (2015) Sustainable Development as a Framework for Ethics and Skills in Higher Education Computing Courses, in: Leal Filho W, Brandli L, Kuznetsova O, Paço A (eds) (2015) *Integrative Approaches to Sustainable Development at University Level*, World Sustainability Series, Springer, Cham, available from: https://doi.org/10.1007/978-3-319-10690-8_24
	Software engineering	Across the software engineering curriculum, students are required to consider **inclusivity** as a key concept in all their designs. This includes accessible interface design, and reflection on how all stakeholders would access the technology.

Cross-curricula	Use of high-performing computer	Ensuring that ratings of computers look at FLOPS/watt as well as raw FLOPS, looking at the power figures in 'Top 500', and also stressing 'Green 500'.
	Infrastructure	Knowledge about environmental impact related to production of silicon devices and disposal of obsolete computing equipment.
	Machine learning/ big data	Knowledge of the power consumption of computer hardware, especially big data processing and blockchain.
	Software engineering	Skills in designing and writing code using good practice to ensure that it is readable, updatable and maintainable (and ideally modular). This reduces total development costs and extends the lifetime of software, making the whole process more sustainable.
	Software engineering	Students are introduced to accessible design principles in first year, then expected to apply them in every subsequent design project (particularly focusing on whether any design decisions would limit access for any stakeholder groups).
Indirectly	Introductory programming	Frame CS1-style/introductory programming module(s) around sustainability-themed project. Jeffrey A Stone and Laura Cruz (2020) Integrative learning in CS1: programming, sustainability, and reflective writing, *Journal of Computing Sciences in Colleges*, vol 35, 8, pp 44–54. https://sites.psu.edu/sustainabilitycis/
	Projects	Projects that use one or more of the UN's Sustainable Development Goals as the focus of the project. Can also involve external partners and the development of proof-of-concept prototypes.
Specific module	Business computing	The module explores ICT use and implementation in areas such as education, health, e-government and environmental sustainability within the context of the UNs Sustainable Development Goals. Students critically appraise the range of ICT issues within a developing country, chosen by themselves from the OECD official development assistance list. They research issues that affect successful implementation of ICTs and give recommendations on the effective implementation of ICTs and how certain Sustainable Development Goals can be achieved through use of ICTs.
Placements		During their placements, students highlight the role that a company's conventional

		practices (such as agile business processes) can actually play in enhancing its sustainability performance, and they also highlight the commercial benefits of environmental and social practices (such as energy and resource efficiency).
Specific module	Computing - Systems and systems thinking	Students are required to read up on Sustainable Development Goals and select their own case study to base a systems thinking analysis on. The output requires presentation of causal maps, a rich picture, and iceberg model and an impact gaps canvas (a 'map the system' tool) as the basis for the systems thinking investigation of a Sustainable Development Goals related problem.

Appendix 3 - Foundations of computer science courses

The term computing applies to an increasingly diverse set of degree courses based on the foundations of computer science. There is a rich set of aspects associated with the computing discipline, including (but not restricted to) the following.

Foundational issues

- theoretical considerations intended to ensure a sound logical basis for the discipline; complexity issues which address feasibility and efficiency concerns; the existence of formal aspects which facilitate automation
- principles of programming languages, compilers and programming environments
- the concept of the algorithm, the concept of a pattern, and notions of reuse
- ideas of abstraction and design, applied in the context of the domain knowledge associated with particular applications and linked to problem-solving
- life cycle and process concepts
- professional, legal, social, cultural and ethical concerns.

Major technologies

- techniques associated with software construction and development, including the development of sociotechnical systems
- electronic/chip design and system-level integration, including bio-inspired developments
- computing systems, including multicore processors and their exploitation, parallel and vector processing systems, distributed systems, cloud computing, quantum computing and grid computing
- pervasive computing, including networks, the internet, mobile computing systems, cybersecurity and social networking systems; the interface with telecommunications and the exploitation of modern communication systems
- methods and techniques for information management, based around sound principles for updating and maintaining information
- appropriate awareness of techniques to address concerns for security, integrity and safety
- artificial intelligence, machine learning, probabilistic programming and its application to small and large data sets
- creative technologies such as virtual and augmented realities.

Discipline areas

This Statement identifies computer science, computer engineering, software engineering, information technology, data science and information systems as discipline areas and outlines the content covered by these. A UK Computing degree may include subject matter from more than one discipline area.

Computer science provides the necessary knowledge to understand and build computational systems. Its main characteristics include:

- the user experience: embracing matters such as digital media, usability in its broadest sense, personalised systems, concern for users with some form of disability, and generally applications of ubiquitous and ambient computing and their effects on user environments and behaviour
- fundamental computational concepts and algorithmic thinking, including recursive, distributed and parallel possibilities and attention to the benefits and the limitations

of these; the role of these in devising approaches to areas of system design, problem-solving, artificial intelligence, simulation and computational modelling
- recognition of the relationships between the concepts of requirements, specification, design, programme and data (in all its forms) validation and maintenance, as well as the power of transformation and proof, and the place of these in computing
- understanding the power behind abstraction, the potential of multiple levels of abstraction and the role this plays in computing
- understanding the opportunities for and the potential of automation, but also the proper balance between automation and how humans effectively interact with computers, recognising the role of redundancy, diversity and separation of concerns in achieving reliable, usable and secure systems, often in the presence of uncertainty
- recognising simplicity and elegance as useful concepts and principles.

Generally, these are expressed in the ability to specify, design and write computer programmes.

The discipline areas of computer engineering, software engineering, information systems, information technology, data science and cybersecurity draw upon the fundamentals of computer science and each other.

Computer engineering is concerned with the realisation of computer science fundamentals in computer hardware. It includes:

- scientific and engineering principles that underpin the design and operation of modern computer hardware and electro-mechanical interfaces
- the understanding of the trade-offs between hardware and software in overall system design
- memory, processors, peripherals, communication and networking
- real-time and embedded systems, mobile devices.

Generally, these are expressed in the ability to understand the construction of, and make best use of, computational devices, interfaces and protocols.

Software engineering is concerned with the building of software systems. It includes:

- problem definition, specification (including formal specification), design, implementation (including debugging) and maintenance, software testing, change management and documentation
- cybersecurity, including information security, and safety-critical systems
- understanding risk, reliability and scalability of the range of possible options and an appreciation of design trade-offs.

Generally, these are expressed in the ability to create fit-for-purpose software in a variety of application domains.

Information technology is concerned with the application of computing technologies to other domains. It includes:

- the selection and application of software and hardware
- integration of components to provide solutions in a variety of application domains
- risk, cybersecurity and service management aspects of IT systems.

Generally, these are expressed in the ability to deliver a computer-based system as a solution to desired needs.

Information systems is concerned with the modelling, codification and storage of data and information for later retrieval and analysis in ways that support decision making. It includes:

- data management, databases, information modelling, indexing and searching
- systems analysis, system life cycle and interactions between information systems and other socio-technical systems, including societal and environmental issues.

Generally, these are expressed in the ability to construct systems that acquire, codify, store, transform and transmit information.

Data science provides a balance between science, statistics and technology. It includes:

- the selection and application of analytical software tools
- the application of probabilistic machine learning techniques
- applying tools and knowledge to address the challenges of small and large data sets.

Cybersecurity involves technology, people, information and processes to enable assured operations in the context of adversaries. It includes:

- the creation, operation, analysis and testing of secure computer systems
- aspects of law, policy, human factors, ethics and risk management.

Degrees in computing, more commonly those at master's level, may be designed to cover a particular specialism or subdiscipline within computing in greater detail, for example, computer graphics, information management, e-commerce, digital media, communications and networking, computing systems architectures, the internet, web science, mobile computing, data warehousing, artificial intelligence, machine learning, medical computing, software project management and the user experience.

ACM Computing Curricula

In the decades since the 1960s, roughly every 10 years, ACM has described curriculum recommendations to the rapidly changing landscape of computer technology. Copies are available at: www.acm.org/education/curricula-recommendations

Computing Curricula – The Overview Report
CC2020: Computing Curricula 2020: Paradigms for Global Computing Education

Computer science
CS2013: Curriculum Guidelines for Undergraduate Programs in Computer Science

Computer engineering
CE2016: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering

Cybersecurity
CSEC2017: Curriculum Guidelines for Post-Secondary Degree Programs in Cybersecurity

Data Science
CCDS2021: Computing Competencies for Undergraduate Data Science Curricula

Information systems
IS2020 Curriculum Update: A Competency Model for Undergraduate Programs in Information Systems

Information Technology
IT2017: Curriculum Guidelines for Baccalaureate Degree Programs in Information Technology.

Software engineering
SE2014 Curriculum Guidelines for Undergraduate Degree Programs in Software Engineering

Fifth edition

Published - 30 March 2022

Southgate House, Southgate Street, Gloucester GL1 1UB
Registered charity numbers 1062746 and SC037786
www.qaa.ac.uk