



Subject Benchmark Statement

Computing: Draft for consultation

June 2015

Contents page

How can I use this document?	1
About Subject Benchmark Statements	2
About this Subject Benchmark Statement	4
1 Introduction	7
2 Nature and extent of computing	8
3 Subject-specific and generic skills.....	11
4 Teaching, learning and assessment.....	13
5 Computing degrees as preparation for professional practice.....	14
6 Benchmark standards	15
Appendix 1: Reference to other curriculum documents	17
Appendix 2: Membership of the review group for the Subject Benchmark Statement for computing	18

How can I use this document?

This document is a 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.

You may want to read this document if you are:

- involved in the design, delivery and review of programmes of study in computing, or related subjects
- a prospective student thinking about studying computing or a current student of the subject, to find out what may be involved
- an employer, to find out about the knowledge and skills generally expected of a graduate in computing.

Explanations of unfamiliar terms used in this Subject Benchmark Statement can be found in the Quality Assurance Agency for Higher Education's (QAA's) glossary.¹

¹ The QAA glossary is available at: www.qaa.ac.uk/about-us/glossary.

About Subject Benchmark Statements

Subject Benchmark Statements form part of the UK Quality Code for Higher Education (Quality Code) which sets out the Expectations that all providers of UK higher education reviewed by QAA are required to meet.² They are a component of Part A: Setting and Maintaining Academic Standards, which includes the Expectation that higher education providers 'consider and take account of relevant Subject Benchmark Statements' in order to secure threshold academic standards.³

Subject Benchmark Statements describe the nature of study and the academic standards expected of graduates in specific subject areas, and in respect of particular qualifications. They provide a picture of what graduates in a particular subject might reasonably be expected to know, do and understand at the end of their programme of study.

Subject Benchmark Statements are used as reference points in the design, delivery and review of academic programmes. They provide general guidance for articulating the learning outcomes associated with the programme, 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 programme design within a framework agreed by the subject community. Further guidance about programme design, development and approval, learning and teaching, assessment of students, and programme monitoring and review is available in Part B: Assuring and Enhancing Academic Quality of the Quality Code in the following Chapters:⁴

- *Chapter B1: Programme Design, Development and Approval*
- *Chapter B3: Learning and Teaching*
- *Chapter B6: Assessment of Students and the Recognition of Prior Learning*
- *Chapter B8: Programme Monitoring and Review.*

For some subject areas, higher education providers may need to consider other reference points in addition to the Subject Benchmark Statement in designing, delivering and reviewing programmes. These may include requirements set out by professional, statutory and regulatory bodies, national occupational standards and industry or employer expectations. In such cases, the Subject Benchmark Statement may provide additional guidance around academic standards not covered by these requirements.⁵ The relationship between academic and professional or regulatory requirements is made clear within individual Statements, but it is the responsibility of individual higher education providers to decide how they use this information. The responsibility for academic standards remains with the higher education provider who awards the degree.

Subject Benchmark Statements are written and maintained by subject specialists drawn from and acting on behalf of the subject community. The process is facilitated by QAA. In order to ensure the continuing currency of Subject Benchmark Statements, QAA initiates regular reviews of their content, five years after first publication, and every seven years subsequently.

² The Quality Code, available at www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code, aligns with the *Standards and Guidelines for Quality Assurance in the European Higher Education Area*, available at: www.enqa.eu/wp-content/uploads/2013/06/ESG_3edition-2.pdf.

³ Part A: Setting and Maintaining Academic Standards, available at: www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/quality-code-part-a.

⁴ Individual Chapters are available at: www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/quality-code-part-b.

⁵ See further Part A: Setting and Maintaining Academic Standards, available at: www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/quality-code-part-a.

Relationship to legislation

Higher education providers are responsible for meeting the requirements of legislation and any other regulatory requirements placed upon them, for example, by funding bodies. The Quality Code does not interpret legislation nor does it incorporate statutory or regulatory requirements. Sources of information about other requirements and examples of guidance and good practice are signposted within the Subject Benchmark Statement where appropriate. Higher education providers are responsible for how they use these resources.⁶

Equality and diversity

The Quality Code embeds consideration of equality and diversity matters throughout. Promoting equality involves treating everyone with equal dignity and worth, while also raising aspirations and supporting achievement for people with diverse requirements, entitlements and backgrounds. An inclusive environment for learning anticipates the varied requirements of learners, and aims to ensure that all students have equal access to educational opportunities. Higher education providers, staff and students all have a role in, and responsibility for, promoting equality.

Equality of opportunity involves enabling access for people who have differing individual requirements as well as eliminating arbitrary and unnecessary barriers to learning. In addition, disabled students and non-disabled students are offered learning opportunities that are equally accessible to them, by means of inclusive design wherever possible and by means of reasonable individual adjustments wherever necessary.

⁶ See further the *UK Quality Code for Higher Education: General Introduction*, available at: www.qaa.ac.uk/publications/information-and-guidance/publication?PubID=181.

About this Subject Benchmark Statement

This Subject Benchmark Statement refers to bachelor's degrees with honours in Computing.⁷

This version of the Statement forms its third edition, following initial publication in 2000 and review and revision in 2007.⁸

Note on alignment with higher education sector coding systems

Programmes of study which use this Subject Benchmark Statement as a reference point are generally classified under the following codes in the Joint Academic Coding System (JACS):⁹

I100	Computer science
I110	Computer architectures & operating systems
I111	Computer architectures
I112	Operating systems
I113	Displays & imaging
I114	High end computing
I115	Parallel computing
I120	Networks & communications
I130	Computational science foundations
I140	Human-computer interaction
I150	Multimedia computing science
I160	Internet
I161	e-business
I190	Computer science not elsewhere classified
I200	Information systems
I210	Information modelling
I220	Systems design methodologies
I230	Systems analysis & design
I240	Databases
I250	Systems auditing
I260	Data management
I270	Intelligent & expert systems
I290	Systems analysis & design not elsewhere classified
I300	Software engineering
I310	Software design
I320	Programming
I321	Procedural programming
I322	Object-oriented programming
I323	Declarative programming

⁷ Bachelor's degrees are at level 6 in *The Framework for Higher Education Qualifications in England, Wales and Northern Ireland* and level 10 in *The Framework for Qualifications of Higher Education Institutions in Scotland*, as published in *The Frameworks for Higher Education Qualifications of UK Degree-Awarding Bodies*, available at: www.qaa.ac.uk/assuring-standards-and-quality/the-quality-code/qualifications.

⁸ Further information is available in the *Recognition Scheme for Subject Benchmark Statements*, available at: www.qaa.ac.uk/publications/information-and-guidance/publication?PubID=190.

⁹ Further information about JACS is available at: www.hesa.ac.uk/content/view/1776/649.

I390	Software engineering not elsewhere classified
I400	Artificial intelligence
I410	Speech & natural language processing
I420	Knowledge representation
I430	Neural computing
I440	Computer vision
I450	Cognitive modelling
I460	Machine learning
I461	Automated reasoning
I490	Artificial intelligence not elsewhere classified
I500	Health informatics
I510	Health technologies
I520	Bioinformatics
I530	Tele healthcare
I590	Health informatics not elsewhere classified
I600	Games
I610	Computer games programming
I620	Computer games design
I630	Computer games graphics
I700	Computer generated visual & audio effects
I710	Computer generated imagery
I900	Others in Computer sciences
I990	Computer sciences not elsewhere classified

Summary of changes from the previous Subject Benchmark Statement (2007)

The following provides an overview of the principal changes to the statement and why they were considered necessary.

- As with the previous review, the 2015 QAA review group identified that the number and type of computing degree programmes continues to proliferate: this has become a cause of confusion for industry and government. Degree programmes entitled Computer Science, Software Engineering, Information Technology and Information Systems, now exist along with more focused degrees such as Computer Games, Computer Networking, Digital Forensics and Multimedia.
- Driven by the expansion of technology, computing is not a static subject. This puts demands on the body of knowledge to expand to include new subject matter and new practices. These two features have led to a widespread debate of the essential content of a degree programme in computing. In order to provide a structure to support this debate and identify groups of related degrees and still permit higher education providers to develop programmes to their own skills and specifications, this Subject Benchmark Statement explicitly identifies a set of discipline areas each building upon the core discipline of Computer Science. These reflect similar areas identified by other subject bodies and are detailed in section 2 (Nature and scope of computing) of this statement.

In revising this Subject Benchmark Statement, a number of changes have been made.

- In line with the revised approach to the diversity of computing programmes, outlined above, the prescriptive body of knowledge has been replaced by reference to the regularly updated Association of Computing Machinery curricula documents (Appendix A).
- With regard to Section 4: Teaching, learning and assessment, a quantity of material of general educational value, but with limited specific relationship to computing, was removed.
- Contextual updates were put in place, including recognition of computing being adopted as a subject within the National Curriculum, the current debates on Computing Graduate Employability, and the 2015 governmental review of accreditation of computing programmes.
- A number of minor edits to improve clarity and readability.

1 Introduction

1.1 In as much as human ingenuity and creativity has fostered the rapid development of the discipline of computing in the past, programmes in computing should not limit those who will lead the development of the discipline in the future.

1.2 Computing is concerned with the understanding, design and exploitation of computation and computer technology - one of the most significant advances of the twentieth and twenty-first centuries. 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, secure reliable and usable systems that support organisational goals.
- helps individuals in their everyday lives.
- is pervasive, ubiquitous and diversely applied to a range of applications, and important components are often invisible to the naked eye.

1.3 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. Whatever the perspective, computing can claim characteristics that, while present in other disciplines, are rarely present in such quantities and combinations.

1.4 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.

1.5 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.

1.6 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 other Subject Benchmark Statements, to capture fully the specific character of their particular degree programmes.

1.7 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. The curriculum documents from the Association of Computing Machinery (ACM) are widely used as a source of guidance on possible curriculum content.

1.8 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 real world and provide early exposure to professional competences.

2 Nature and extent of computing

2.1 The computing discipline is evolving at a rapid rate, touching upon all aspects of life. Computing delivers innovative solutions to problems, and drives technological, economic and social progress.

2.2 Computing as a discipline consists of central elements: mathematics; fundamentals of computation; and realisation of computer systems in both hardware and software.

2.3 Computing graduates apply their understanding, skills, knowledge and experience to create social and economic value by building secure, reliable and usable systems.

2.4 Computing includes aspects that overlap with areas of interest to a number of adjacent subjects. Examples of such areas are:

- engineering, especially parts of electrical and electronic engineering
- physics, with concern for multimedia and device-level development of computing components
- mathematics (logic and theoretical models of computation)
- business (information services)
- philosophy, physiology and psychology (human computer interaction and aspects of artificial intelligence)
- linguistics.

2.5 The application of computational techniques across science and engineering has fundamentally affected practices within those disciplines. Therefore computing is both a rigorous academic discipline in its own right and also facilitates and supports a wide range of other disciplines, from computational physics to computational social science.

2.6 The concept of computational thinking is central to the discipline. 'Computational thinking is using abstraction and decomposition when attacking a large complex task or designing a large complex system. It is separation of concerns. It is choosing an appropriate representation for a problem or modeling the relevant aspects of a problem to make it tractable. It is using invariants to describe a system's behaviour succinctly and declaratively' J M Wing (2008) Computational thinking and thinking about computing, [*Philosophical Transactions A*](#)

2.7 The term computing applies to an increasingly diverse set of degree programmes all based on the foundations of Computer Science. This Statement identifies Computer Science, Computer Engineering, Software Engineering, Information Technology, 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.

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

- 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 modelling
- recognition of the relationships between the concepts of specification, programme and data (in all its forms), 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.

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

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

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

- memory, processors, peripherals, communication and networking;
- real-time systems and mobile devices.

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

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

- problem definition, specification (including formal specification), design, implementation and maintenance, software testing
- security and safety-critical systems
- the understanding of the range of possible options and an appreciation of design trade offs.

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

2.12 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.

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

2.13 Information Systems is concerned with the modelling, codification and storage of data and information for later retrieval and analysis. It includes:

- data management, databases, information modelling, indexing and searching.

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

2.14 The graduates from such programmes will typically tend to seek opportunities in the discipline area they studied.

2.15 Many higher education providers deliver degrees focused on specific aspects of computing in society, for example computer networking, games, multimedia, health informatics. These programmes 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.

2.16 Additionally, computing is widely taught in joint and interdisciplinary programmes 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 programmes.

2.17 This Statement does not specify syllabi or include a body of knowledge. The ACM, in conjunction with the Institute of Electrical and Electronics Engineers (IEEE) and other professional societies, maintain (and regularly update) curricula in several areas: Computer Science, Computer Engineering, Information Systems, Information Technology and Software Engineering. These documents should be used to inform programme design and curriculum content. (See Appendix 1).

2.18 The title of a programme cannot describe the whole of its content. However, the programme title must not be divorced from graduate knowledge, skills and abilities. There are natural overlaps between the different identified discipline areas. So 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 of the degree programme should reference the curriculum from the dominant discipline area.

2.19 Programme designers, students and employers will need to be aware of this spectrum of programme identities, but behind such variation there are key ideas that can be expected to characterise any honours degree programme in computing:

- the concept of computational thinking, the recognition of its main elements and the relevance of these to everyday life
- the computing system (including an information system), and the process of developing or analysing it is important; understanding of the system and its operation will go deeper than a mere external appreciation of what the system does or the way(s) in which it is used
- the balance of practice and theory such that practical activity is supported by an understanding of underlying principles.

3 Subject-specific and generic skills

3.1 Students of computing are expected to develop a wide range of skills. These may be divided into three broad categories:

- Computing related cognitive skills
- Computing related practical skills
- generic skills for employability.

3.2 Cognitive, practical and generic skills need to be placed in the context of the programme of study as designed by the higher education provider and/or the possible pathways selected by the individual student. There is an implicit interplay between these identified skills within and across these three categories. The extent to which students acquire these skills depends on the emphasis of individual degree programmes.

3.3 Computing related cognitive skills.

- Computational thinking including its relevance to everyday life.
- Knowledge and understanding: demonstrate knowledge and understanding of essential facts, concepts, principles and theories relating to computing and computer applications as appropriate to the programme of study.
- Modelling: use such knowledge and understanding in the modelling and design of computer-based systems for the purposes of comprehension, communication, prediction and the understanding of trade offs.
- Requirements, practical constraints and computer-based systems (and this includes computer systems, information, security, embedded, and distributed systems) in their context: recognise and analyse criteria and specifications appropriate to specific problems, and plan strategies for their solutions.
- Critical evaluation and testing: analyse the extent to which a computer-based system meets the criteria defined for its current use and future development.
- Methods and tools: deploy appropriate theory, practices and tools for the specification, design, implementation and evaluation of computer-based systems.
- Professional considerations: recognise the professional, economic, social, environmental, moral and ethical issues involved in the sustainable exploitation of computer technology and be guided by the adoption of appropriate professional, ethical and legal practices.

3.4 Computing-related practical skills.

- The ability to specify, design and construct reliable, secure and usable computer-based systems.
- The ability to evaluate systems in terms of quality attributes and possible trade offs presented within the given problem.
- The ability to plan and manage projects to deliver computing systems within constraints of requirements, timescale and budget.
- The ability to recognise any risks and safety aspects that may be involved in the deployment of computing systems within a given context.
- The ability to deploy effectively the tools used for the construction and documentation of computer applications, with particular emphasis on understanding the whole process involved in the effective deployment of computers to solve practical problems.
- The ability to critically evaluate and analyse complex problems, including those with incomplete information, and devise appropriate solutions.

3.5 Generic Skills for Employability. Students are expected to develop a wide range of generic skills to ensure they become effective in the workplace, to the benefit of themselves, their employer and the wider economy. Students who develop generic skills, and are able to evidence and demonstrate such skills, will gain significant advantage when seeking employment. It is the responsibility of higher education providers to provide every student the opportunity to acquire and evidence generic skills; it is the responsibility of the student to make the most of that opportunity.

3.6 Intellectual skills: critical thinking; making a case; numeracy and literacy; information literacy. The ability to construct well argued and grammatically correct documents. The ability to locate and retrieve relevant ideas, and ensure these are correctly and accurately referenced and attributed.

3.7 Self-management: self-awareness and reflection; goal setting and action planning; independence and adaptability; acting on initiative; innovation and creativity. The ability to work unsupervised, plan effectively and meet deadlines, and respond readily to changing situations and priorities.

3.8 Interaction: reflection and communication: 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).

3.9 Team working and management: the ability to recognise and make best use of the skills and knowledge of individuals to collaborate. To be able to identify problems and desired outcomes and negotiate to mutually acceptable conclusions. To understand the role of a leader in setting direction and taking responsibility for actions and decisions.

3.10 Contextual awareness: the ability to understand and meet the needs of individuals, business and the community, and to understand how workplaces and organisations are governed.

4 Teaching, learning and assessment

4.1 Computing programmes deploy a diverse range of learning, teaching and assessment methods to enhance and reinforce the student learning experience. This diversity of practice is a strength of the subject. Whichever methods are employed, strategies for teaching, learning and assessment deliver opportunities for the achievement of the learning outcomes, demonstrate the attainment of learning outcomes and recognise the range of student backgrounds.

4.2 Curriculum design is informed by current developments, reflecting appropriate research, scholarship, industrial and business practices, together with an understanding of potential graduate destinations. Students achieve an understanding of computing through significant exposure to practical coursework and substantial individual and group-project work. Project types include design-and-build, consultancy and research-led, which develop both independence of thought and the ability to work effectively in a team. Teaching and learning needs to be placed within the context of social, ethical, legal, professional and economic factors relevant to computing.

4.3 The following aspects of curriculum delivery have particular relevance to computing degree programmes:

- encouraging students to reflect, evaluate, select, justify, communicate and be innovative in their problem solving
- provision for the development of a range of personal and generic skills
- a major activity allowing students to demonstrate ability in applying practical and analytical skills (as they are present in the programme as a whole): this will often take the form of a project carried out in the final year
- computing-related case studies employed to indicate the application of student learning after graduation.

4.4 All forms of work-based learning, including activities such as industrial placements, are seen as a valued part of a programme and are properly integrated in terms of preparation of students before this activity, debriefing, building on the experience afterwards and assessment.

4.5 High-quality software, tools and materials condition the expectation of students and their approach to practice, and this is an essential dimension of the educational process. Access to software and communications facilities enables students to extend their horizons. Apart from the usual range of languages, basic software including software libraries, graphics packages, computer-aided software engineering (CASE) tools, integrated development environments, theorem provers, project management software, and planning systems, as appropriate to the programme of study, might be provided.

5 Computing degrees as preparation for professional practice

5.1 There are many different types of computing degree programme, but all are designed to equip their graduates with knowledge, understanding and skills which will enable them to begin a professional career in some aspect of computing. The possession of a computing degree is seen by many computing employers as an essential indication that these attributes have been achieved. Successful graduates from computing programmes are highly sought after.

5.2 Not all graduates will proceed with a professional career in computing. The attributes of computing graduates also make them attractive to many non-computing employers: manufacturing, finance, consultancy, public services, creative industries and the arts, as well as entrepreneurs in their own right.

5.3 Within the context of teaching computing to honours degree level, graduates act as the agents to transfer technology and best-practice when they ultimately move into employment.

5.4 Computing degrees in the UK may be accredited by UK professional bodies (BCS, The Chartered Institute for IT, and the Institute for Engineering and Technology (IET)) as fulfilling (or partially fulfilling) criteria for professional standing. Such bodies take their own view of the amount of exemption a programme delivers to students graduating from it.

6 Benchmark standards

6.1 Benchmark standards are defined at threshold and typical levels for bachelor's degrees and further comments are made with reference to integrated master's degrees.

The threshold level

6.2 Set here at the bottom of the honours class would be treated by many higher education providers as disappointing performance, given the entry qualifications of their students, and it is not the outcome expected of them. But since the students are graduating with an honours degree, which is itself a significant level of educational achievement, this Subject Benchmark Statement tries to set out positively what minimally acceptable graduates are able to do.

6.3 On graduating with an honours degree in Computing, students should be able to:

- demonstrate a requisite understanding of the main body of knowledge for their programme of study
- 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
- produce work involving problem identification, the analysis, design and development of a system with accompanying documentation, recognising the important relationships between these stages
- the work will show problem-solving and evaluation skills drawing on supporting evidence
- demonstrate generic skills, an ability to work under guidance and as a team member.
- identify appropriate practices within a professional, legal and ethical framework and understand the need for continuing professional development.

The typical level

6.4 On graduating with an honours degree in computing, students should be able to:

- demonstrate a sound understanding of the main areas of the body of knowledge within their programme of study, with an ability to exercise critical judgement
- critically analyse and apply essential concepts, principles and practices of the subject in the context of loosely defined scenarios, showing effective judgement in the selection and use of tools and techniques
- produce work involving problem identification, the analysis, the design or the development of a system, with appropriate documentation, recognising the important relationships between these. The work will show problem solving and evaluation skills, draw upon supporting evidence and demonstrate a good understanding of the need for a high quality solution
- demonstrate generic skills with an ability to show organised work both as an individual and as a team member and with minimum guidance
- apply appropriate practices within a professional, legal and ethical framework and identify mechanisms for continuing professional development and lifelong learning.

Excellence

6.5 While the Subject Benchmark Standards on page 13 are defined for threshold and typical levels, programmes in computing will provide opportunities for students to achieve far more highly.

6.6 Such students:

- will be able to contribute significantly to the analysis, design or the development of systems which are complex and fit for purpose, recognising the important relationships between these
- will be creative and innovative in their application of the principles covered in the curriculum
- will be able to exercise critical evaluation and review of both their own work and the work of others.

Master's level

6.7 Integrated master's degrees (MComp, MEng and MSci) include the outcomes of bachelor's degrees with honours and go beyond them to provide a greater range and depth of specialist knowledge, often within a research and industrial environment, as well as a broader and more general academic base. Such programmes provide a foundation for leadership.

6.8 Integrated master's programmes of study are designed as an integrated whole from entry to completion, although earlier parts may be delivered in common with a parallel bachelor's degree with honours.

Appendix 1: Reference to other curriculum documents

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

Computer Science

Computer Science 2013: Curriculum Guidelines for Undergraduate Programs in Computer Science

Computer Engineering

CE 2004: Curriculum Guidelines for Undergraduate Degree Programs in Computer Engineering (at the time of writing this is under revision)

Information Systems

IS 2010 Curriculum Update: The Curriculum Guidelines for Undergraduate Degree Programs in Information Systems

Information Technology

IT 2008: The Computing Curricula Information Technology

Software Engineering

GSwE2009: Curriculum Guidelines for Graduate Degree Programs in Software Engineering

Computing Curricula 2005: The Overview Report

CC 2005 provides undergraduate curriculum guidelines for five defined disciplinary areas of computing: Computer Science, Computer Engineering, Information Systems, Information Technology and Software Engineering.

Appendix 2: Membership of the review group for the Subject Benchmark Statement for computing

Membership of the review group for the 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

Employer representative

Robert Koger	Vision Semantics and IET Academic Accreditation Committee
--------------	-----------------------------------------------------------

Student reader

Emilia Todorova	Glasgow Caledonian University
-----------------	-------------------------------

Membership of the review group for the 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 review 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	University of Keele
Dr PC Capon	University of Manchester
Mr G Davies	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)
Mr P McGrath
Dr A Norman
Mr SJ Oldfield
Ms A Rapley
Professor VJ Rayward-Smith

University of Strathclyde
Leeds Metropolitan University
University of Cambridge
University of Plymouth
Kingston University
University of East Anglia

QAA1202 - June 2015

© The Quality Assurance Agency for Higher Education 2015
Southgate House, Southgate Street, Gloucester GL1 1UB

Tel: 01452 557 000
Email: enquiries@qaa.ac.uk
Website: www.qaa.ac.uk

Registered charity numbers 1062746 and SC037786