UKForCE
UK Forum for Computing Education

The new computing curriculum

Terminology and definitions

UK Forum for Computing Education, June 2016

1 Motivation

The new English computing curriculum is described by the statutory programmes of study. This two-page curriculum describes ten years of study, so it is necessarily dense.

This paper explains some of the language used in the new computing curriculum, and gives pointers to where to find out more.

A caution. People disagree about definitions. In the end it is not very fruitful to argue about what a particular term means; there is typically no "right answer". It is more fruitful to ask what we want our students to learn.

2 Computer science, information technology, and digital literacy

The Royal Society Report "Shut down or restart: the way forward for computing in schools" identified three strands that should form part of the computing curriculum:

- Computer science
- Information technology
- Digital literacy

The programmes of study deliberately does not use this taxonomy, preferring to regard the subject as an integrated whole. However, the categories can be useful as a checklist to help evaluate whether a particular scheme of work is covering all the bases.

The general term "**computing**" is *the term used for the whole subject* area in the English national curriculum. In particular, it is clearly broader than computer science, including information technology and digital literacy.

With that in mind, here is a bit more detail.

2.1 Computer science: foundations

Here is a short definition of computer science

> " Computer science is the study of information and computation. "

The introduction of computer science into the school curriculum, from primary school onwards, is the defining characteristic of the new computing curriculum. The first of the four aims of the programmes of study is:

> " The national curriculum aims to ensure that can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation. "

The Royal Society report defined computer science like this: *"computer science should be interpreted as referring to the scientific discipline of Computer Science, covering principles such as algorithms, data structures, programming, systems architecture, design, problem solving etc."*

In more detail:

- Computer science is a foundational subject discipline, like mathematics or natural science, that every child should have the opportunity to learn.
- A "subject discipline" is characterised by ideas, theories, methods, principles, and knowledge that continue to be useful for a lifetime; these ideas and principles should survive successive waves of technology.

- Physics studies physical objects and their interaction; computer science studies *computation* and *information*. For example, DNA encodes information about our genome in sequences of base pairs (information); while the biochemical pathways that govern the expression of those genes are essentially interacting computational processes. Or, in the context of programming, Wirth coined the equation algorithms + data = programs.
- Computer science in schools includes the foundational concepts of computational thinking such as algorithms, abstraction and logical reasoning, the practical applications of these through programming, as well as an understanding of how particular technologies (such as processors, storage and computer networks) work.

2.2 Information technology: applications

The Royal Society report defines information technology (IT) like this:

> " Information Technology is the assembly, deployment, and configuration of digital systems to meet user needs "

The third aim of the programmes of study echoes this language

> " The national curriculum aims to ensure that can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems. "

Note that

- Information Technology is more applied than Computer Science, as the use of "Technology" in its name implies.
- But it still has substantial intellectual content, including information theory, databases, networks, web applications and services, information security, and systems design.

Unlike computer science, the government decided not to establish a GCSE in Information Technology, so beyond Key Stage 3 these topics are examined in non-GCSE qualifications.

Even with the new emphasis on computer science in the national curriculum, it remains essential that pupils leave school able to accomplish useful work using a variety of software applications across a range of digital devices: the abilities to collect and analyse data and create high quality content using computers across a range of media are as important as ever, as are the knowledge and understanding which underpin these.

2.3 Digital literacy: implications

The Royal Society report defined digital literacy like this:

> " Digital literacy is the basic skill or ability to use a computer confidently, safely and effectively. "

So in other words:

- Digital literacy is analogous to numeracy, and literacy in reading or writing: these are basic skills without which people are severely handicapped in navigating the modern world.
- Most students should be digitally literate (in this sense) by the time they leave primary school, just as they should be numerate and be able to read and write. If not, remedial work is needed.
- Digital literacy skills continue to be improved, through application and practice, at secondary school. They are best assimilated if they are practiced in a disciplinary context (topic/practice)
- Digital literacy certainly embraces e-safety.

This is a minimum baseline. The programmes of study goes a little further, saying “Computing also ensures that pupils become digitally literate – able to use, and express themselves and develop their ideas through, information and communication technology – at a level suitable for the future workplace and as active participants in a digital world.” Shortly after, the fourth of the four aims of the curriculum is

> “ The national curriculum aims to ensure that all pupils are responsible, competent, confident, and creative users of information and communication technology. ”

Other work has advocated a much broader and more ambitious definition of “digital literacy”. A good exemplar is the 2010 Futurelab report “It’s not chalk and talk any more: school approaches to developing digital literacies”. Another helpful source is the JISC Guide to Developing students’ digital literacy.

So there is a spectrum of meanings for “digital literacy”, but all are focused around confident and creative use of digital technology. The broader the range of a student’s experience, and the more opportunity for creative and critical use of technology they have, the better.

Schools have a statutory duty to ensure their pupils welfare is safeguarded and promoted. The national curriculum adds to this requirement that they be taught to ‘use technology safely, respectfully, responsibly and securely, including protecting their online identity and privacy’. Whilst online safety is best addressed through whole school approaches, a broad and balanced computing education must ensure that pupils consider the implications of technology in their lives and for society as a whole.

3 Computational thinking

Much writing about the new computing curriculum mentions **computational thinking**. But what exactly is “computational thinking”? It is a term that has broad resonance, but which is surprisingly hard to pin down. Jeannette Wing’s original definition is this

> “ Computational thinking is the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively carried out by an information-processing agent. ”

Here is another attempt, from the CAS Curriculum for Computer Science:

> “ Computational thinking is the process of recognising aspects of computation in the world that surrounds us, and applying tools and techniques from computing to understand and reason about both natural and artificial systems and processes. ”

Rather than getting hung up on a precise definition, it is more helpful to recognise *characteristics* of computational thinking, which you can see in both these definitions.

- Computational thinking is something that *people* do, not something that *computers* do. It includes the ability to think logically, algorithmically and (at higher levels) recursively and abstractly.
- You can’t write a program without engaging in computational thinking, but computational thinking has far broader applications than programming. Thinking about how to get all the children out of school fastest when there is a fire alarm is a problem where computational thinking would help, even though it may never be expressed in a computer program.
- That is why Wing uses the clunky term “information processing agent”. A person is an information processing agent. So is a bumble bee. So is a cell. So is a computer. They all do things (agent) in systematic way, in response to information that they receive.
- Computational thinking is not an abstract intellectual skill; it’s about getting things done in the world that surround us. Hence “…effectively carried out by…” and “…applying tools and techniques…”.

- Computational thinking gives an understanding of the *artificial* world, but it also gives a new way of understanding the *natural world.* For example, cells are driven by DNA, which works very much like a program, written on a “tape” of nucleotides; termites build mounds of astonishing complexity, not driven by a central brain, but by the interaction of lots of tiny termite brains running tiny programs; the global ecosystem is the result of multiple sub-ecosystems all interacting with each other; and so on.
- Insights from computer science are becoming directly useful in understanding the natural order, often thorough simulations that abstract the key pieces of a complex biological system, and implement their behaviour in an algorithmic way. Computation and information processing take place in many more types of system than traditional ‘computers’.

There is a collection of pointers to resources supporting computational thinking on this CAS Community resource.

Computational thinking can be a useful term, because it suggests (rightly) that the new curriculum is not simply about teaching children about computers, or about some exotic academic subject. Rather it gives students systematic thinking skills that will be useful for the rest of their lives, both in grappling with technology, and more broadly.

Computational thinking is not everything! A well-rounded student of computing will also be proficient in other generic skills and processes, including: thinking critically, reflecting on one’s work and that of others, communicating effectively both orally and in writing, being a responsible user of computers, and contributing actively to society.

4 Programming and coding

Programming is explicitly brought out in the second aim of the computing programme of study:

> “ The national curriculum aims to ensure that all pupils can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems. ”

There is much talk of coding and programming, to the extent that you might think that ‘coding’ pretty much *was* the new part of the computing curriculum.

Absolutely not! Programming plays the same role in computer science that practical investigations do in maths or science. Programming animates and incarnates the subject, and brings computer science to life; it is creative, and engaging. It illustrates otherwise-abstract concepts in completely concrete terms. It is also an incredibly useful skill.

Programming should start with computational thinking. In order to program a computer to do something, it’s necessary to understand the something, to break this down into parts, to think of how these will be represented in a computer and to consider the steps or rules that the computer must follow in order to do the thing; typically, it’s only then that this solution can be implemented in a particular programming language as code. Think perhaps of programming as the expression of algorithms as code.

Computer science is more than programming, just as chemistry is more than Bunsen burners and test tubes.

One of the main ways in which the new curriculum could fail would be if the reality on the ground turned out to be that schools just teach coding as a useful and engaging life skill.

5 Further reading

- Decoding the new computing programmes of study is an informal guide by Simon Peyton Jones, who chaired the working group that drafted the curriculum. Its goal is to “unpack” the dense language of the curriculum, and explain more of the intent that lies behind it.
- The original CAS curriculum for computer science gives a detailed exposition of the case for computer science in the school curriculum, and its content.