

# Curriculum design

## The approach

### Coherence and flexibility

The Teach Computing Curriculum is structured in units. For these units to be coherent, the lessons within a unit must be taught in order. However, across a year group, the units themselves do not need to be taught in order, with the exception of 'Programming' units, where concepts and skills rely on prior learning and experiences.

### Knowledge organisation

The Teach Computing Curriculum uses the National Centre for Computing Education's computing taxonomy to ensure comprehensive coverage of the subject. This has been developed through a thorough review of the KS1-4 computing programme of study, and the GCSE and A level computer science specifications across all awarding bodies. All learning outcomes can be described through a high-level taxonomy of ten strands, ordered alphabetically as follows:

- **Algorithms** – Be able to comprehend, design, create and evaluate algorithms
- **Computer networks** – Understand how networks can be used to retrieve and share information, and how they come with associated risks
- **Computer systems** – Understand what a computer is, and how its constituent parts function together as a whole
- **Creating media** – Select and create a range of media including text, images, sounds and video
- **Data and information** – Understand how data is stored, organised, and used to represent real-world artefacts and scenarios
- **Design and development** – Understand the activities involved in planning, creating, and evaluating computing artefacts
- **Effective use of tools** – Use software tools to support computing work
- **Impact of technology** – Understand how individuals, systems, and society as a whole interact with computer systems
- **Programming** – Create software to allow computers to solve problems
- **Safety and security** – Understand risks when using technology, and how to protect individuals and systems

The taxonomy provides categories and an organised view of content to encapsulate the discipline of computing. Whilst all strands are present at all phases, they are not always taught explicitly.

For these units to be coherent, the lessons within a unit must be taught in order. However, across a year group, the units themselves do not need to be taught in order, with the exception of 'Programming' units, where concepts and skills rely on prior learning and experience.

## Spiral curriculum

The units for key stages 1 and 2 are based on a spiral curriculum. This means that each of the themes is revisited regularly (at least once in each year group), and pupils revisit each theme through a new unit that consolidates and builds on prior learning within that theme.

This style of curriculum design reduces the amount of knowledge lost through forgetting, as topics are revisited yearly. It also ensures that connections are made even if different teachers are teaching the units within a theme in consecutive years.

## Physical computing

The Teach Computing Curriculum acknowledges that physical computing plays an important role in modern pedagogical approaches to computing, both as a tool to engage pupils and as a strategy to develop pupils understanding in more creative ways. Additionally, physical computing supports and engages a diverse range of pupils in tangible and challenging tasks.

The physical computing units in key stage 2 are:

- Year 5 – Programming A – Selection in physical computing
- Year 6 – Programming B – Sensing movement

Your local Computing Hub may be able to loan you the kit you need to teach the physical computing units from our curriculum ([nccce.io/hubs](https://nccce.io/hubs)).

## Online safety

The unit overviews for each unit show the links between the content of the lessons and the national curriculum and Education for a Connected World framework ([nccce.io/efacw](https://nccce.io/efacw)). These references have been provided to show where aspects relating to online safety, or digital citizenship, are covered within the Teach Computing curriculum. Not all of the objectives in the Education for a Connected World framework are covered in the Teach Computing curriculum as some are better suited to personal, social, health, and economic (PSHE) education; spiritual, moral, social, and cultural (SMSC) development; and citizenship. However, the coverage required for the computing national curriculum is provided.

Schools should decide for themselves how they will ensure that online safety is being managed effectively in their setting, as the scope of this is much wider than just curriculum content.

# Structure of the units of work

Every unit of work in the Teach Computing Curriculum contains: a unit overview; a learning graph, to show the progression of skills and concepts in a unit; lesson content –including a detailed lesson plan, slides for learners, and all the resources you will need; and formative and summative assessment opportunities.

## Teach Computing Curriculum overview

	Computing Systems and Networks	Creating Media	Programming A	Data and Information	Creating Media	Programming B
Year 3	Connecting computers (3.1)*	Stop-frame animation (3.2)	Sequencing sounds (3.3)	Branching databases (3.4)	Desktop publishing (3.5)	Events and actions in programs (3.6)
Year 4	The Internet (4.1)	Audio production (4.2)	Repetition in shapes (4.3)	Data logging (4.4)	Photo editing (4.5)	Repetition in games (4.6)
Year 5	Systems and searching (5.1)	Video production (5.2)	Selection in physical computing (5.3)	Flat-file databases (5.4)	Introduction to vector graphics (5.5)	Selection in quizzes (5.6)
Year 6	Communication and collaboration (6.1)	Web page creation (6.2)	Variables in games (6.3)	Spreadsheets (6.4)	3D modelling (6.5)	Sensing movement (6.6)

\*The numbers in brackets are a 'quick code' reference for each unit, e.g. 1.3 refers to the third year one unit in the recommended teaching order.

## Unit summaries

	Computing systems and networks	Creating media	Programming A	Data and information	Creating media	Programming B
Year 3	<p><b>Connecting computers</b> Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks</p>	<p><b>Stop-frame animation</b> Capturing and editing digital still images to produce a stop frame animation that tells a story</p>	<p><b>Sequencing sounds</b> Creating sequences in a block-based programming language to make music.</p>	<p><b>Branching databases</b> Building and using branching databases to group objects using yes/no questions.</p>	<p><b>Desktop publishing</b> Creating documents and modifying text, images and page layouts for a specific purpose.</p>	<p><b>Events and actions in programs</b> Writing algorithms and programs that use a range of events to trigger sequences of actions.</p>
Year 4	<p><b>The internet</b> Recognising that the internet is a network of networks including the WWW, and why we should evaluate online content.</p>	<p><b>Audio production</b> Capturing and editing audio to produce a podcast, ensuring that copyright is considered.</p>	<p><b>Repetition in shapes</b> Using a text-based programming language to explore count-controlled loops when drawing shapes.</p>	<p><b>Data logging</b> Recognising how and why data is collected over time, before using data loggers to carry out an investigation,</p>	<p><b>Photo editing</b> Manipulating digital images, and reflecting on the impact of the changes and whether the required purpose is fulfilled,</p>	<p><b>Repetition in games</b> Using a block-based programming language to explore count-controlled and infinite loops when creating a game.</p>

## Unit summaries

	Computing systems and networks	Creating media	Programming A	Data and information	Creating media	Programming B
Year 5	<p><b>Systems and searching</b> Recognising IT systems in the world and how some can enable searching on the internet.</p>	<p><b>Video production</b> Planning, capturing, and editing video to produce a short film.</p>	<p><b>Selection in physical computing</b> Exploring conditions and selection using a programmable microcontroller.</p>	<p><b>Flat-file databases</b> Using a database to order data and create charts to answer questions.</p>	<p><b>Introduction to vector graphics</b> Creating images in a drawing program by using layers and groups of objects.</p>	<p><b>Selection in quizzes</b> Exploring selection in programming to design and code an interactive quiz.</p>
Year 6	<p><b>Communication and collaboration</b> Exploring how data is transferred by working collaboratively online.</p>	<p><b>Webpage creation</b> Designing and creating webpages, giving consideration to copyright, aesthetics and navigation.</p>	<p><b>Variables in games</b> Exploring variables when designing and coding a game.</p>	<p><b>Introduction to spreadsheets</b> Answering questions by using spreadsheets to organise and calculate data.</p>	<p><b>3D modelling</b> Planning, developing, and evaluation 3D computer models of physical objects.</p>	<p><b>Sensing movement</b> Designing and coding a project that captures inputs from physical devices.</p>