



Computing Curriculum Intent

This information outlines the knowledge, language and concepts that should be taught in Hambleton Computing. It includes:

- A summary of the Computing knowledge and principles that underpin our approach
- Long Term Sequence (Curriculum Progression Map) for Computing

It is our intention that through studying Computing, pupils become more expert as they progress through the curriculum, accumulating, connecting and making sense of the substantive knowledge and disciplinary knowledge/skills.

1. **declarative knowledge** (know that) - this is the subject knowledge and explicit vocabulary used to learn about the content. Common misconceptions are explicitly revealed as non-examples and positioned against known and accurate content.
2. **procedural knowledge** (know how) – this is knowing how to collect, use, interpret, understand and evaluate learning through the Computing knowledge that is taught. It is not assumed that pupils will acquire these skills by luck or hope.

We acknowledge that the Computing curriculum is split into three main pillars and content areas:

Computer science Information technology Digital literacy

‘However, these pillars do not sit separately from each other. Knowledge from each pillar complements the others and some subject content only exists at the interplay between these 3 pillars.’ Therefore all learning outcomes are planned out under four strands, which provides categories and an organised view of content to encapsulate the discipline of computing:

Computer Systems and Networks - Understand what a computer is, and how its constituent parts function together as a whole, and understand how networks can be used to retrieve and share information, and how they come with associated risks

Programming - Create software to allow computers to solve problems

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

Embedded within these main strands, are the following Digital Literacy skills that are crucial for pupils use of technology in the world:

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

Safety and security — Understand risks when using technology, and how to protect individuals and systems

Computing is planned so that the retention of knowledge is much more than just 'in the moment knowledge', which can be seen in the way in which we structure our lessons. The cumulative nature of the curriculum is made memorable by the implementation of Bjork's desirable difficulties, including retrieval and spaced retrieval practice, word building and deliberate practice tasks. This powerful interrelationship between structure and research-led practice is designed to increase substantive knowledge and accelerate learning within and between study modules. That means the foundational knowledge of the curriculum is positioned to ease the load on the working memory: new content is connected to prior learning. The effect of this cumulative model supports opportunities for children to associate and connect significant computing concepts, over time, and with increasing expertise and knowledge.

Our Computing curriculum has sequenced the national curriculum into meaningful and connected 'chunks' of content to reduce the load on the working memory as well as creating coherent and strong long-term memories. The sequence of substantive and disciplinary knowledge enables pupils to become 'more expert' with each study and grow an ever broadening and coherent mental model of the subject. This guards against superficial, disconnected and fragmented Computing knowledge and weak disciplinary knowledge. High frequency, multiple meaning words (Tier 2) are taught explicitly and help make sense of subject specific words (Tier 3).

Progression Overview



Early Years

As young children take part in a variety of tasks with digital devices, such as moving a Bee Bot around a classroom, they will already be familiar with the device before being asked to undertake tasks related to the key stage one (KS1 - ages 5 - 7 years) computing curriculum, such as writing and testing a simple program. Not only will children be keen to again use a device they had previously enjoyed using, their cognitive load will also be reduced, meaning they are more likely to succeed when undertaking activities linked to the next stage in their learning. Within the revised EYFS statutory framework, there are opportunities within each area of the framework to enable practitioners to effectively prepare children for studying the computing curriculum.

Implementation

We implement our intent using NCCE Teach Computing programme of study. The Teach Computing Curriculum (ncce.io/tcc) is a comprehensive collection of materials produced to support teaching, facilitating the delivery of the entire English computing curriculum. All content is free, and editable under the Open Government Licence (OGL — ncce.io/ogl), ensuring that the resources can be tailored to each individual teacher and school setting. The materials are suitable for all pupils irrespective of their skills, background, and additional needs. 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. The Teach Computing Curriculum has been designed to reduce teacher workload. To ensure this, the Teach Computing Curriculum includes all the resources a teacher needs, covering every aspect from planning, to progression mapping, to supporting materials.

Inclusive and ambitious

The Teach Computing Curriculum has been written to support all pupils. Each lesson is sequenced so that it builds on the learning from the previous lesson, and where



appropriate, activities are scaffolded so that all pupils can succeed and thrive. Scaffolded activities provide pupils with extra resources, such as visual prompts, to reach the same learning goals as the rest of the class. Exploratory tasks foster a deeper understanding of a concept, encouraging pupils to apply their learning in different contexts and make connections with other learning experiences. As well as scaffolded activities, embedded within the lessons are a range of pedagogical strategies, which support making computing topics more accessible.

Research-informed

The subject of computing is much younger than many other subjects, and as such, there is still a lot more to learn about how to teach it effectively. To ensure that teachers are as prepared as possible, the Teach Computing Curriculum builds on a set of pedagogical principles (see the ‘Pedagogy’ section of this document), which are underpinned by the latest computing research, to demonstrate effective pedagogical strategies throughout. To remain up-to-date as research continues to develop, every aspect of the Teach Computing Curriculum is reviewed each year and changes are made as necessary.

Pedagogy

The National Centre for Computing Education’s pedagogical approach consists of 12 key principles underpinned by research: each principle has been shown to contribute to effective teaching and learning in computing. It is recommended that computing teachers use their professional judgement to review, select, and apply relevant strategies for their pupils. These 12 principles are embodied by the Teach Computing Curriculum, and examples of their application can be found throughout the units of work at every key stage. Beyond delivering these units, you can learn more about these principles and related strategies in the National Centre for Computing Education pedagogy toolkit (ncce.io/pedagogy).

Lead with concepts

Support pupils in the acquisition of knowledge, through the use of key concepts, terms, and vocabulary, providing opportunities to build a shared and consistent understanding. Glossaries, concept maps, and displays, along with regular recall and revision, can support this approach.



Work together

Encourage collaboration, specifically using pair programming and peer instruction, and also structured group tasks. Working together stimulates classroom dialogue, articulation of concepts, and development of shared understanding.

Get hands-on

Use physical computing and making activities that offer tactile and sensory experiences to enhance learning. Combining electronics and programming with arts and crafts (especially through exploratory projects) provides pupils with a creative, engaging context to explore and apply computing concepts.

Model everything

Model processes or practices — everything from debugging code to binary number conversions — using techniques such as worked examples and live coding. Modelling is particularly beneficial to novices, providing scaffolding that can be gradually taken away.

Foster program comprehension

Use a variety of activities to consolidate knowledge and understanding of the function and structure of programs, including debugging, tracing, and Parson's Problems. Regular comprehension activities will help secure understanding and build connections with new knowledge.

Create projects

Use project-based learning activities to provide pupils with the opportunity to apply and consolidate their knowledge and understanding. Design is an important, often overlooked aspect of computing. Pupils can consider how to develop an artefact for a particular user or function and evaluate it against a set of criteria.

Add variety

Provide activities with different levels of direction, scaffolding, and support that promote learning, ranging from highly structured to more exploratory tasks. Adapting your instruction to suit different objectives will help keep all pupils engaged and encourage greater independence.



Challenge misconceptions

Use formative questioning to uncover misconceptions and adapt teaching to address them as they occur. Awareness of common misconceptions alongside discussion, concept mapping, peer instruction, or simple quizzes can help identify areas of confusion.

Make concrete

Bring abstract concepts to life with real-world, contextual examples, and a focus on interdependencies with other curriculum subjects. This can be achieved through the use of unplugged activities, proposing analogies, storytelling around concepts, and finding examples of the concepts in pupils' lives.

Structure lessons

Use supportive frameworks when planning lessons, such as PRIMM (Predict, Run, Investigate, Modify, Make) and (Use-Modify-Create). These frameworks are based on research and ensure that differentiation can be built in at various stages of the lesson.

Read and explore code first

When teaching programming, focus first on code 'reading' activities, before code writing. With both block-based and text-based programming, encourage pupils to review and interpret blocks of code. Research has shown that being able to read, trace, and explain code augments pupils' ability to write code.

Curriculum Sequences

Our computing knowledge and skills are covered through the NCCE programme to ensure that children are taught the appropriate content to support their development. Our teaching of Computing is progressive. The Teach Computing curriculum is a spiral curriculum with units building progressively from year to year. The curriculum is structured into 6 units for each year group, and each unit is broken down into lessons. Below shows the 6 units, which are covered:



Computing Systems and Networks	Creating Media	Creating Media	Data and Information	Programming	Programming
recognise common uses of information technology beyond school understand computer networks including the internet; how they can provide multiple services, such as the world wide web; and the opportunities they offer for communication and collaboration Searching	use technology purposefully to create, organise, store, manipulate and retrieve digital content select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information	use technology purposefully to create, manipulate and retrieve digital content select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals	use technology purposefully to organise and store digital content select, use and combine a variety of software (including internet services) on a range of digital devices to accomplish given goals, including collecting, analysing, evaluating and presenting data and information	understand what algorithms are; how they are implemented as programs on digital devices; and that programs execute by following precise and unambiguous instructions create and debug simple programs use logical reasoning to predict the behaviour of simple programs	design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts use sequence, selection, and repetition in programs; work with variables and various forms of input and output
Online Safety Effective Use of Tools Impact of Technology					

Computer Science

Computer Systems and Networks: The Computer Systems and Network unit build on the prior learning from the year before and is sequenced progressively throughout the primary phase. The graphics below show how the unit spirally progress from year to year.

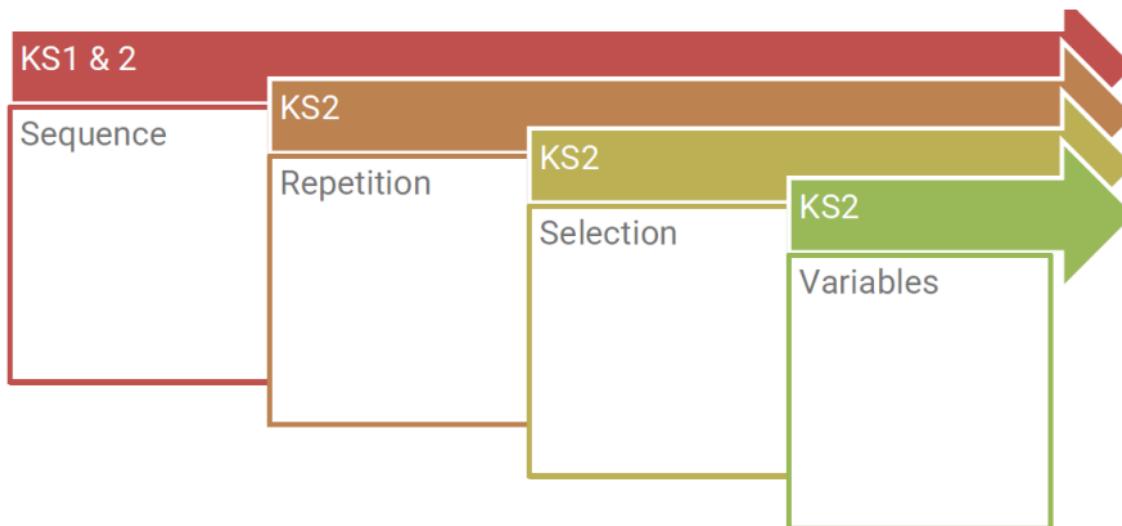
Computing Systems and Networks Strand Overview					
Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Technology Around Us 6 hrs Recognising technology all around us and beginning to use keyboard and mouse.	Information Technology Around Us 6 hrs Recognising that there are peripherals that work with devices.	Connecting Computers 6 hrs Understand the IPO model. Recognise computers are connected within a network	What is the Internet? 6 hrs Recognising the internet as a network of networks Internet fundamentals	Sharing Information 6 hrs How computing systems share information How the internet shares information Collaboration	Communication 6 hrs Search Engines Websites Online Communication

Programming:

Our curriculum incorporates two programming units: Programming A and Programming B, to highlight the emphasis of Computer Science statements in the

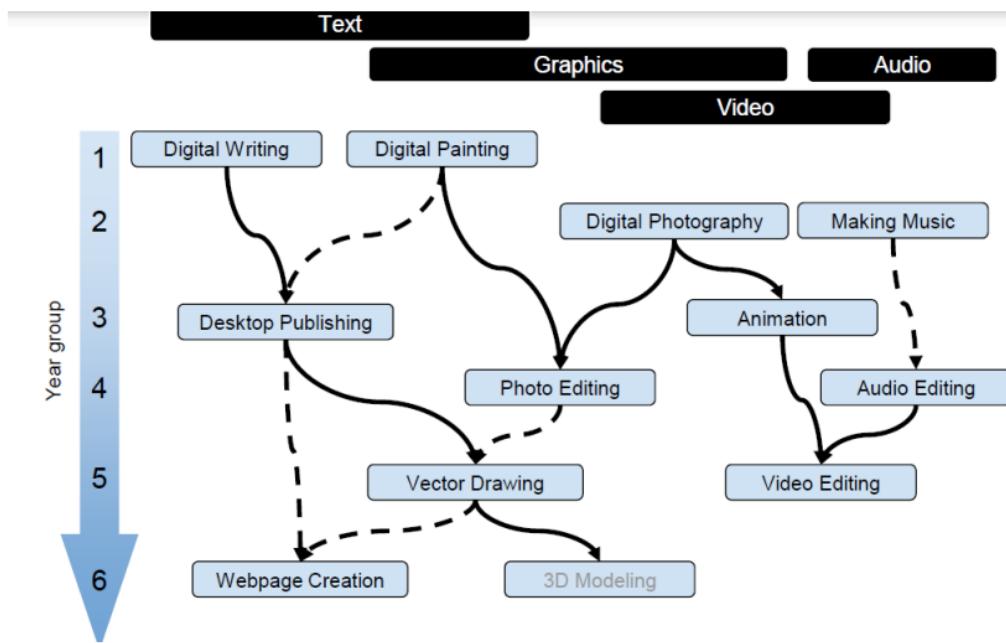


national curriculum. Programming is revisited twice within a year, with unit B progressing from what they have learnt from unit A. Within Key Stage 1, they look at two different formats: Beebots and Scratch Junior. The learning from Year 1 is built upon in Year 2. In Key Stage 2, the national curriculum statement “use sequence, selection and repetition in programmes; work with variables and various forms of input and output” has been identified as a key area, where progression of skills is needed and therefore, the curriculum organisation is sequenced to ensure this. For example, Year 3 focus on sequence, Year 4 focus on repetition, Year 5 on selection and Year 6 on variables. Each year group looks at their skills in depth and masters these over the two unit.



Information Technology

Creating Media Our Creating Media units focus on four main themes: text, graphics, video and audio. Rather than revisiting the same software each year, the curriculum is sequenced to revisit skills learnt prior and apply these using new software, with each year group delivering two units.



Data and Information

The Data and Information unit works in the same way as the Computer Systems and Networks strand, with a unit per year group, looking at different forms of presenting data through Key Stage 2.

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 (ncce.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.

Lesson Structure

Lessons typically are split into six phases:

- **CONNECT**

This provides an opportunity to connect the lesson to prior learning from a previous module or lesson. Teachers return children's attention to the previous lesson's knowledge note/the big idea for the learning module, including key vocabulary.



Examples of thinking harder routines include Flick Back 5, Recap questions, Quizzing. Retrieval practice allows all pupils to take time to remember things and activate their memories. Quizzing allows questions to be asked and allows pupils to carry out retrieval practice. Cumulative quizzing, allows for a few questions to be asked each lesson, which are built upon the previous lesson.

- EXPLAIN

This is the explicit teaching that needs to take place. Teachers should ensure they are clear what they want children to know and remember. They plan for and explicitly address common misconceptions so they can address these in lessons as they arise. They should be clear about the substantive knowledge and the vocabulary that they want children to understand in the session. This can be developed using key information, facts, and images so that explanations are precise.

- EXAMPLE

Providing pupils with high-quality examples is essential for learning. Pupils need to see worked examples. My turn, our turn, your turn is a technique that can be used to explicitly teach vocabulary and new concepts. Prepared examples should be carefully planned and need to be evident in teaching. An example in geography could be demonstrating how to label a map, before labelling a map together.

- ATTEMPT

Guiding pupil practice allows pupils to rehearse, rephrase and elaborate their learning. Children need the chance to attempt and verbalise their understanding. Children's own attempts are what help them to secure their understanding. Children need to have time to struggle and understand for themselves. This is not necessarily something that is recorded in books. This phase provides opportunities for teachers to check in with pupils to see who may need more challenge/support/scaffolds and if any misconceptions have arisen that need to be addressed. Extending the previous geography example, pupils could practice labelling a map.

- APPLY



This is where pupils would typically begin to record in individual pupil activity sheets, on Seesaw or in floor books in KS1 or on digital floor books in KS2. The number of scaffolds may vary.

- **CHALLENGE**

Teachers get the children to interrogate their learning - summarise, explain, compare and contrast. Tools are built into routines to reduce overload and allow for hard thinking. These can be adapted for children based on their individual needs.

Impact

In order to identify the impact our curriculum is having on our pupils, we check the extent to which learning has become permanently embedded in children's long-term memory in addition to looking for excellence in their outcomes. We use four main tools to quality assure the implementation and impact of our curriculum:

- Learning observations/through drop ins help to evaluate subject knowledge, explanations, expectations, opportunities to learn, pupil responses, participation and relationships.
- Professional growth models help to improve staff subject knowledge and evidence informed practice such as retrieval and spaced practice, interleaving and explicit instruction techniques.
- Assessment and achievement articulate the outcomes from tasks and tests, how well the content is understood and what the strengths and limitations are; it informs what to do next.
- Pupil Book Studies help to evaluate curriculum structures, teaching methods, pupil participation and response through a dialogic model.

When undertaking these we ask the following key questions:

How well do pupils remember the content that they have been taught?

Do books and pupil discussions radiate excellence?

Does learning 'travel' with pupils and can they deliberately reuse it in more sophisticated contexts?



Teachers employ a range of strategies both at and after the point of teaching to check the impact of their teaching on the permanence of pupils' learning. These include: retrieval practice, vocabulary use and application, deliberate practice and rephrasing of taught content, cumulative quizzing within the learning sequence, summarising and explaining the learning question from the sequence, tests and quizzes. Teachers use information from tasks, tests, pupil book studies and other monitoring to support learning by responding to the gap between where pupils are and where they need to be. In lessons, they adapt explanations and examples to address misconceptions and provide additional practice or challenge where required. After lessons or tests, they analyse pupils' responses to identify shared and individual gaps in learning and misconceptions. Teachers then adjust subsequent planned teaching in response.

We use summative assessment (Hambleton Non-Core Assessment document) is 'to provide an accurate shared meaning without becoming the model for every classroom activity' (Christodolou, 2017). If our curriculum is effective, it will lead to improvements in summative assessments over time. Teacher assessment judgements are against an agreed assessment model (the curriculum).

Pupil book study is used as a method to quality assure our curriculum by talking to the children and looking in pupils' books. We do this after content has been taught to see the extent to which pupils are knowing more, remembering more and able to do more. In preparation, we review the planned content, knowledge and vocabulary, so that conversations with pupils are meaningful and focused on what has been taught. When looking at books, we look at the content and knowledge, teaching sequence and vocabulary. We also consider pupils' participation and consider the explanations and models used, the tasks the pupils are asked to do, the ability to answer carefully selected questions and retrieve information and the impact of written feedback. We ask careful questions that probe their knowledge, understanding and skills.

The Subject Leader undertakes a range of activities to understand what the curriculum looks like across the school and how well pupils know more, remember more and can do more as a result. In addition to the above tools, they use learning walks, planning reviews and book looks. They use their findings to support teachers



to improve how they implement subjects and to make recommendations about the suitability of the intent for their subject.