**Computing Curriculum**

**The Teach Computing Scheme of Work**

The Teach Computing Curriculum is a comprehensive collection of materials produced to support the teaching and delivery of the entire Computing curriculum for KS1 & 2. The Teach Computing Curriculum was created by the Raspberry Pi Foundation on behalf of the National Centre for Computing Education (NCCE).

The aims of the Teach Computing Curriculum are as follows:

* Show the breadth and depth of the computing curriculum, particularly beyond programming and
* Demonstrate how computing can be taught well, based on research.

The Teach Computing Curriculum resources are regularly updated. 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 and slides for learners and
* formative and summative assessment opportunities.

**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. 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

**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 approaches in 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 the Teach Computing Curriculum are:

**Year 5** – selection in physical computing, which uses

a Crumble controller and

**Year 6** – sensing, which uses a microbit.

The Computing Hub/Durham County can loan the equipment needed to teach the physical computing units from this curriculum.

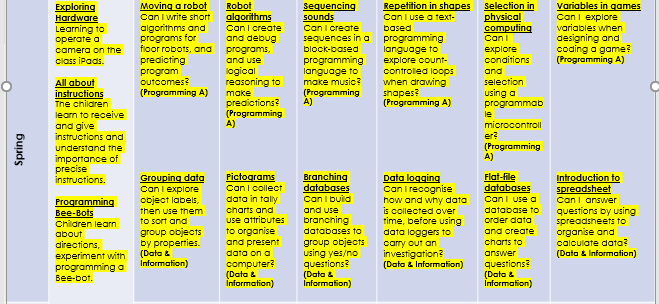
**Core Principles**

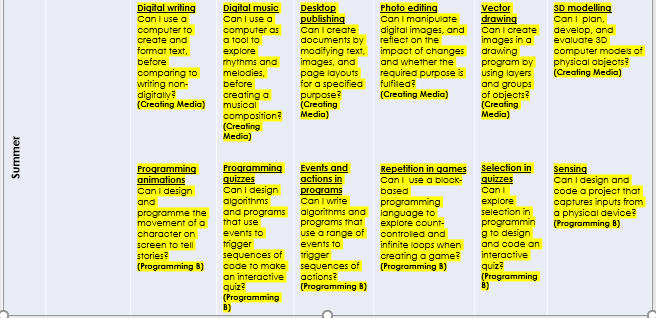
**Inclusive and ambitious**

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.

**Computing Long Term Plan**







**Units of Work**

Each unit of work contains:

Lesson plan;

Lesson PowerPoint;

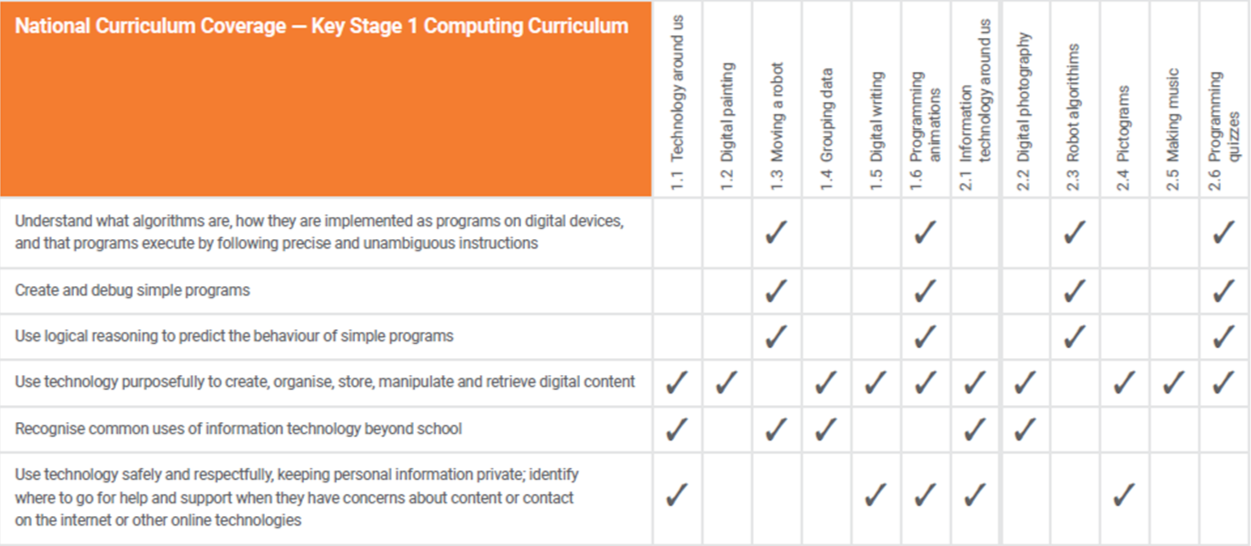
Lesson Worksheets;

Progression Learning Graph;

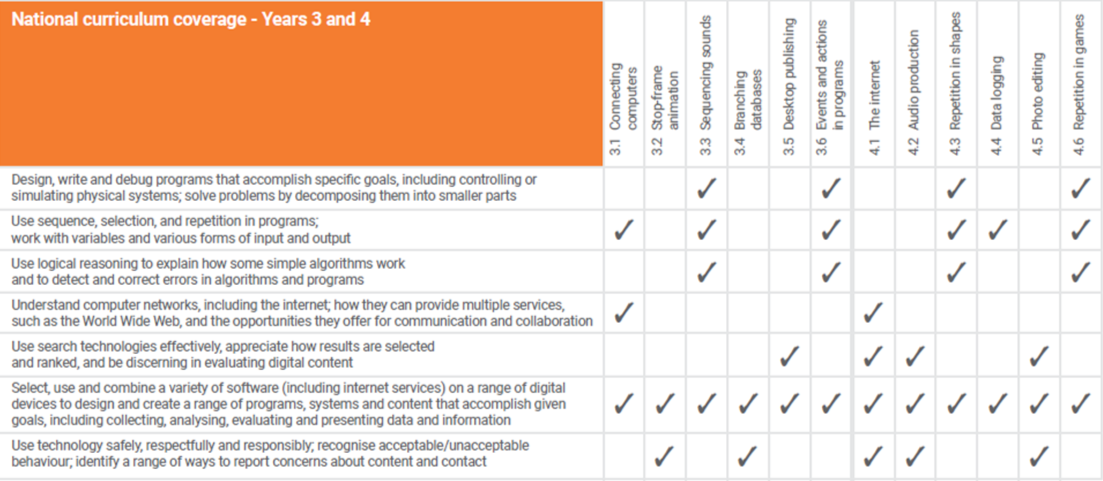
Summative Assessment & answers and

Unit Overviews.

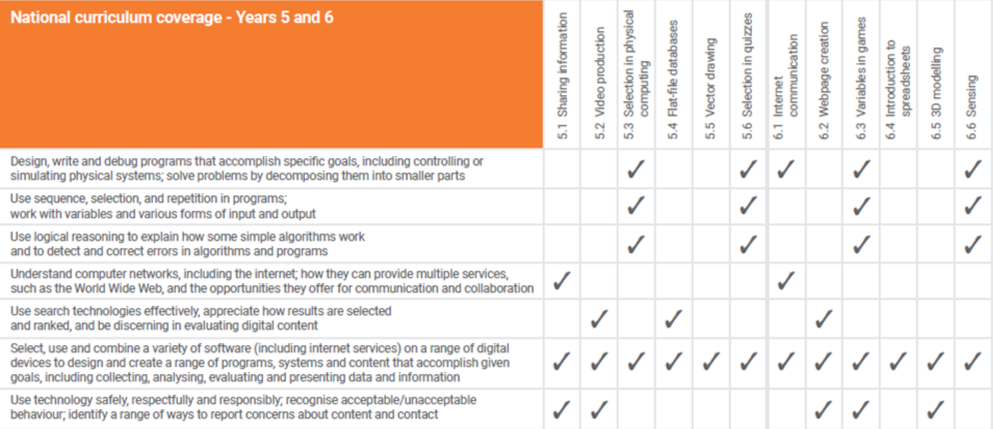
**KS1 National Curriculum Coverage**



**Year 3 & 4 National Curriculum Coverage**



**Year 5 & 6 National Curriculum Coverage**



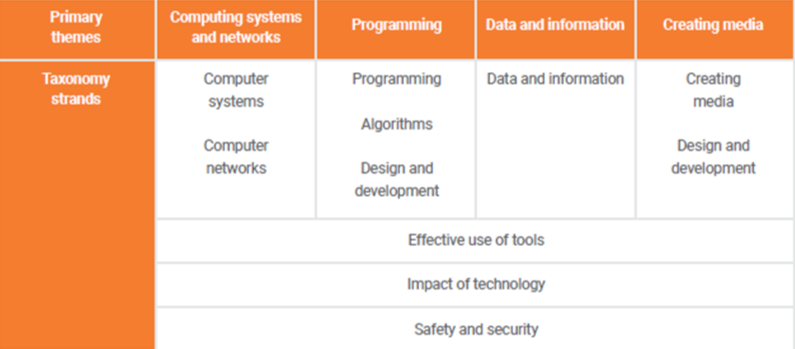
**Progression**

**Progression across key stages**

All learning objectives have been mapped to the National Centre for Computing Education’s taxonomy of ten strands, which ensures that units build on each other from one key stage to the next.

**Progression across year groups**

Within the Teach Computing Curriculum, every year group learns through units within the same four themes, which combine the ten strands of the National Centre for Computing Education’s taxonomy. This approach allows the spiral curriculum approach to progress skills and concepts from one-year group to the next.



**Progression within a Unit — Learning Graphs**

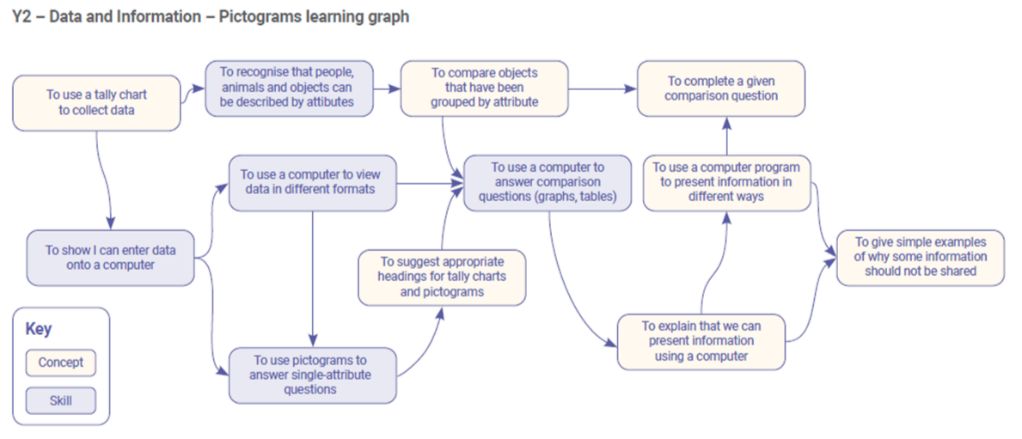
Learning graphs are provided as part of each unit and demonstrate progression through concepts and skills. In order to learn some of those concepts and skills, pupils need prior knowledge of others, so the learning graphs show which concepts and skills need to be taught first and which could be taught at a different time.

The learning graphs often show more statements than there are learning objectives. All of the skills and concepts learnt are included in the learning graphs. Some of these skills and concepts are milestones, which form learning objectives, while others are smaller steps towards these milestones, which form success criteria.

The wording of the statements may be different in the learning graphs than in the lessons, as the learning graphs are designed for teachers, whereas the learning objectives and success criteria are age-appropriate so that they can be understood by pupils.

**Progression Learning Graph Example**

Used to introduce unit and ticked off as and when completed



**Assessment**

**Adapting to our Setting**

As there are no nationally agreed levels of assessment, the assessment materials are designed to be used and adapted in a way that best suits our needs.

The summative assessment materials inform teacher judgements around what a pupil has understood in each computing unit, and could feed into a school’s assessment process, to align with their approach to assessment in other foundation subjects.

**Formative Assessment KS1 & 2**

Every **lesson** includes formative assessment opportunities. These opportunities are listed in the **lesson plan** and are included to ensure that misconceptions are recognised and addressed if they occur. They vary from teacher observation or questioning, to marked activities.

These assessments are vital to adapt teaching to suit the needs of the pupils, to change parts of the lesson, such as how much time is spent on a specific activity, in response to these assessments.

The learning objective and success criteria are introduced in the slides at the beginning of every lesson. At the end of every lesson, pupils are invited to assess how well they feel they have met the learning objective using thumbs up, thumbs sideways, or thumbs down **(Green/Yellow/Red marks on their work!)**

This gives pupils a reminder of the content that has been covered, as well as a chance to reflect. It is also a chance to see how confident the class is feeling so that changes can be made to subsequent lessons.

**KS1 Summative Assessment**

When we assess, we want to ensure that we are assessing a pupil’s understanding of computing concepts and skills, as opposed to their reading and writing skills. Therefore, we encourage **observational** **assessment** while pupils are still developing their literacy skills. We believe that this is the most reliable way to capture an accurate picture of learning.

**Observing Learning**

To capture summative assessment data of KS1 pupils, we use the success criteria in each lesson and capture some of the following while the lesson is taking place:

The work that pupils complete (marking);

Notes on conversations or discussions that you have or hear during an activity;

Photographs of the work that pupils produce during an activity and

The pupils’ self-assessments at the end of the lesson.

This data is to support teachers’ assessments of the pupils’ understanding of the concepts and skills that were taught in the lesson. To make these assessments, we may also use one, or a combination of, the following strategies:

* Focussing on different pupils each lesson;
* Creating checklists of what you expect to see and
* Focussing on specific pupils.

**End of the Unit**

A pupil working at age-related expectations should be able to meet the success criteria for each lesson by the end of the unit. However, some pupils may take longer to grasp certain skills and concepts and therefore may achieve a success criterion from a lesson at a later date.

At the end of a unit, any observations made across each of the lessons can help to determine an overall snapshot of a pupil’s understanding of the content from that unit. A summative assessment excel document is completed for each child where gradings of WT/AT and GD given for each lesson and their success criteria. A best fit level can then be made for the end of each unit. This excel document follows each child throughout their school life.

**KS2 Summative Assessment**

Every unit includes an optional summative assessment framework in the form of either a **multiple-choice quiz (MCQ)** or a **rubric.** All units are designed to cover both skills and concepts from across the computing national curriculum. Units that focus more on conceptual development include an MCQ. Units that focus more

on skills development end with a project and include a rubric. However, within the ‘Programming’ units, the assessment framework (MCQ or rubric) has been selected on a best-fit basis.

**Multiple Choice Quiz (MCQ)**

Each of the MCQ questions has been carefully chosen to represent learning that should have been achieved within the unit. In writing the MCQs, a diagnostic assessment approach is used to ensure that the assessment of the unit is useful to determine both how well pupils have understood the content, and what

pupils have misunderstood and if they have not achieved as expected.

Each MCQ includes an answer sheet that highlights the misconceptions that pupils may have if they have chosen a wrong answer. This ensures that teachers know which areas to return to in later units.

**Rubric**

The rubric is a tool to help assess **project-based work**. Each rubric covers the application of skills that have been directly taught across the unit, and highlights whether the pupil is approaching (emerging), achieving (expected), or exceeding the expectations for their age group. Pupils’ projects can be assessed, focussing on the appropriate application of computing skills and concepts.

**End of the Unit**

A pupil working at age-related expectations should be able to meet the success criteria for each lesson by the end of the unit. However, some pupils may take longer to grasp certain skills and concepts and therefore may achieve a success criterion from a lesson at a later date.

At the end of a unit, any observations made across each of the lessons can help to determine an overall snapshot of a pupil’s understanding of the content from that unit. A summative assessment excel document is completed for each child where gradings of WT/AT and GD given for each lesson and their success criteria. A best fit level can then be made for the end of each unit. This excel document follows each child throughout their school life.

**Class Computing Folders and Class Computing Book**

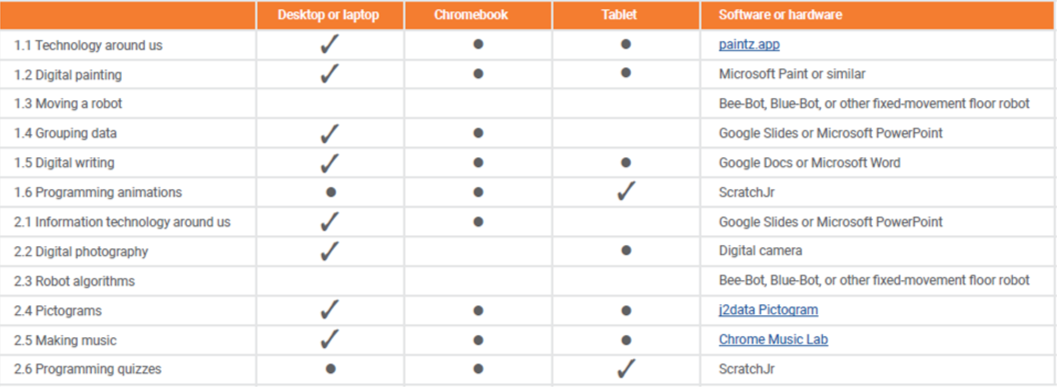
There are a lot of sheets involved with the Teach Computing Scheme across the year so manilla folders are used to keep pupil’s work together. A Class Computing Book is also kept to celebrate and evidence different levels of pupil’s work.

**Software and Hardware**

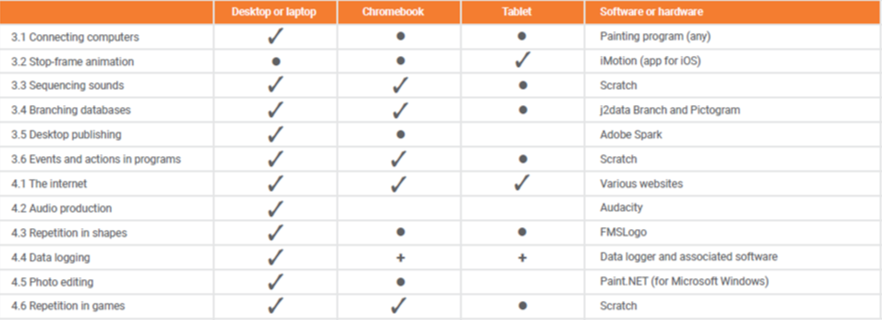
Computing is intrinsically linked to technology and therefore requires that pupils experience and use a range of digital tools and devices. Careful consideration has been given to the hardware and software selected for the units. The primary consideration was how a tool would best allow pupils to meet learning objectives; the learning always came first and the tool second.

To make the units of work more accessible to pupils and teachers, the following software and hardware overviews are to be used. The overviews should not be seen as an explicit requirement for schools. Schools can use alternative tools that offer the same features as described in the units. All of the learning objectives can be met with alternative hardware and software, as the learning objectives are not designed to be tool-specific.

**KS1 Software and Hardware Overview**



**LKS2 Software and Hardware Overview**



**UKS2 Software and Hardware Overview**

