



## Computing long-term plan (based on Teach Computing) 2021-2022

### Key Stage 2

Overchurch Junior School Key Elements: **To Connect To Communicate To Code To Collect**

The approach Overchurch Junior School is one of Coherence and flexibility. We use the The Teach Computing Curriculum structured in units and taught in blocks as well as cross curricular. The units are taught in order to be coherent. 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. 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. We teach a sequenced, coherent computing curriculum, designed to prepare children for KS3 and for a digital, fast moving world.

#### **KS 2 NATIONAL CURRICULUM:**

- 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
- use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs
- 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
- use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating 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 safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact

Year	Autumn		Spring		Summer	
<b>3</b>	<b><u>COMPUTING SYSTEMS AND NETWORKS</u></b>	<b><u>CREATING MEDIA ANIMATION</u></b>	<b><u>DESKTOP PUBLISHING</u></b>	<b><u>BRANCHING DATABASES</u></b>	<b><u>SEQUENCE IN MUSIC EVENTS AND ACTIONS</u></b>	<b><u>PRORAMMING B EVENTS AND ACTIONS</u></b>
	Connecting computers Identifying that digital devices have inputs, processes, and outputs, and how devices can be connected to make networks	Stop-frame animation Capturing and editing digital still images to produce a stop-frame animation that tells a story	Creating documents by modifying text, images, and page layouts for a specified purpose	Branching databases Building and using branching databases to group objects using yes/no questions	Children will use Scratch. The final product will be to make a piano	Children programme a sprite within a maze using Code Studio

4	<p><b><u>COMPUTING SYSTEMS THE INTERNET</u></b></p> <p>The internet Recognising the internet as a network of networks including the WWW, and why we should evaluate online content.</p>	<p><b><u>AUDIO EDITING</u></b></p> <p>Capturing and editing audio to produce a podcast, ensuring that copyright is considered</p>	<p><b><u>PHOTO EDITING</u></b></p> <p>Manipulating digital images, and reflecting on the impact of changes and whether the required purpose is fulfilled.</p>	<p><b><u>DATA LOGGING</u></b></p> <p>Using a text-based programming language to explore count-controlled loops when drawing shapes.</p>	<p><b><u>REPETITION IN SHAPES</u></b></p> <p>Recognising how and why data is collected over time, before using data loggers to carry out an investigation</p>	<p><b><u>REPETITION IN GAMES</u></b></p> <p>Using a block-based programming language to explore count-controlled and infinite loops when creating a game.</p> <p>Course D Studiocode.org</p>
5	<p><b><u>COMPUTING NETWORKS SHARING INFORMATION</u></b></p> <p>Sharing information Identifying and exploring how information is shared between digital systems.</p>	<p><b><u>CREATING MEDIA VECTOR DRAWING</u></b></p> <p>Creating images in a drawing program by using layers and groups of objects. Course E Studiocode.org</p>	<p><b><u>CREATING MEDIA VIDEO EDITING</u></b></p> <p>Planning, capturing, and editing video to produce a short film.</p>	<p><b><u>FLAT FILE DATABASES</u></b></p> <p>Flat-file databases Using a database to order data and create charts to answer questions</p>	<p><b><u>SELECTION IN PHYSICAL COMPUTING</u></b></p> <p>Exploring conditions and selection using a programmable microcontroller</p>	<p><b><u>SELECTION IN QUIZZES</u></b></p> <p>Exploring selection in programming to design and code an interactive quiz.</p>
6	<p><b><u>COMPUTING SYSTEMS AND NETWORKS INTERNET COMMUNICATION</u></b></p> <p>Recognising how the WWW as a communication tool Learning how search engines work Evaluate different methods of communication</p>	<p><b><u>3D MODELLING</u></b></p> <p>Planning, developing, and evaluating 3D computer models of physical objects.</p>	<p><b><u>WEBPAGE CREATION</u></b></p> <p>Designing and creating webpages, considering copyright, aesthetics, and navigation. Studio Code</p>	<p><b><u>SPREAD SHEETS</u></b></p> <p>Answering questions by using spreadsheets to organise and calculate data.</p>	<p><b><u>VARIABLES IN GAMES</u></b></p> <p>Exploring variables when designing and coding a game. Studio Code Course F Studiocode.org</p>	<p><b><u>SENSING</u></b></p> <p>Designing and coding a project that captures inputs from a physical device.</p>

The Primary National Curriculum for Computing is split into three strands: information technology, digital literacy and computer science. At Overchurch Junior School, our Computing curriculum strives to create excitement, creativity and overall love of technology through using various elements and skills to code, connect, communicate and collect.

At Overchurch Junior School, we follow the Teach Computing Curriculum which is a spiral curriculum. This means that each of the themes is revisited regularly (at least once in each year group), and children 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 ensuring that connections are made.

**To Code:** Being able to code and program through the use of computer science helps children of all ages to understand how computers and networks function. It gives all children the opportunity to learn basic computer programming, create on-screen computer games and write algorithms that program a range of devices.

**To Connect:** Being able to connect through digital literacy is vital in the development of a child's understanding on how to be safe and responsible whilst using technology. Children focus on how to keep safe online.

**To Communicate:** Being able to communicate in a variety of ways through the use of digital literacy, by developing keyboard skills and confidence whilst using a range of media programs. Children will learn how to safely collect pictures, videos and data and use appropriate applications and programmes to display, present or organise their information.

Please see: <https://teachcomputing.org/curriculum/key-stage-2> for more information about our Computing curriculum.

Lessons also include regular teaching of e-safety to ensure that children feel confident when using computers and the Internet, and know what to do if they come across something either inappropriate or uncomfortable.