Learning in Computing Skills Progression

Aims: The National Curriculum for Computing aims to ensure that all pupils:
- *can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation;*
- *can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems;*
- *can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems; and*
- *are responsible, competent, confident and creative users of information and communication technology.*

Intent

We offer a structured sequence of lessons, helping teachers to ensure that they have covered the skills required to meet the aims of the national curriculum. The content allows for a broad, deep understanding of computing and how it links to children's lives. It offers a range of opportunities for consolidation, challenge and variety. This allows children to apply the fundamental principles and concepts of computer science. They develop analytical problem-solving skills and learn to evaluate and apply information technology. It also enables them to become responsible, competent, confident and creative users of information technology.
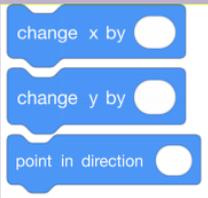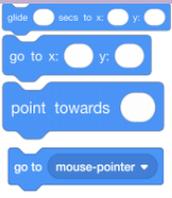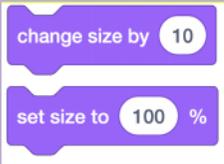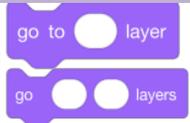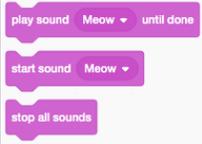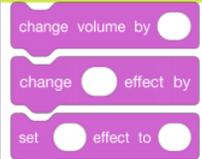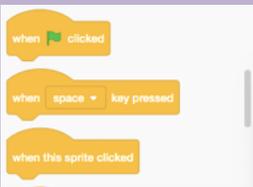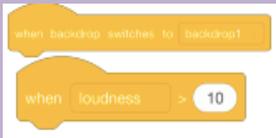
Implementation

Each lesson contains revision, analysis and problem-solving. Through the sequence of lessons, we intend to inspire pupils to develop a love of the digital world, see its place in their future and give teachers confidence. Cross-curricular links are also important in supporting other areas of learning. Our lesson plans and resources help children to build on prior knowledge at the same time as introducing new skills and challenges. In KS1, the focus is on developing the use of algorithms, programming and how technology can be used safely and purposefully. In KS2, lessons still focus on algorithms, programming and coding but in a more complex way and for different purposes. Children also develop their knowledge of computer networks, internet services and the safe and purposeful use of the internet and technology. Data Handling is featured more heavily in UKS2. Skills learnt through KS1 and LKS2 are used to support data presentation.

Impact

Learning in computing will be enjoyed across the school. Teachers will have high expectations and quality evidence will be presented in a variety of forms. Children will use digital and technological vocabulary accurately, alongside a progression in their technical skills. They will be confident using a range of software. Children will see the digital world as part of their world, extending beyond school, and understand that they have choices to make. They will be confident and respectful digital citizens going on to lead happy and healthy digital lives.

	Y1	Y2	Y3	Y4	Y5	Y6
NC Programme of Study	understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions		design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts			
	create and debug simple programs		use sequence, selection, and repetition in programs; work with variables and various forms of input and output			
	use logical reasoning to predict the behaviour of simple programs		use logical reasoning to explain how some simple algorithms work and to detect and correct errors in algorithms and programs			
	use technology purposefully to create, organise, store, manipulate and retrieve digital content		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			
	recognise common uses of information technology beyond school		use search technologies effectively, appreciate how results are selected and ranked, and be discerning in evaluating digital content			
	use technology safely and respectfully, keeping personal information private; identify where to go for help and support when they have concerns about content or contact on the internet or other online technologies		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			

	Y1	Y2	Y3	Y4	Y5	Y6
To code (using Scratch) **Computer Science:** Pupils need to understand what algorithms are – this is the basis of what they need to know in order to write computer programs. Pupils need to be able to write algorithms and programs. They also need to be able to find mistakes (bugs) and fix them. When children write programs they will learn that there are often different ways of getting the right outcome, and they need to be able to evaluate the programs to decide which is the most efficient.	**Motion** Specify the number of steps to travel, direction and turn (using beebots) to use directional language to control Beebot.	**Motion** Specify the number of steps to travel, direction and turn (using Scratch)	**Motion** Use specified co-ordinates to control movement	**Motion** Use specified co-ordinates to control movement	**Motion** Specify types of rotation giving the number of degrees	**Motion** Set IF conditions for movements.
	Looks Add text strings, show and hide objects and change features. Capture an image and transfer to another device or application – Purple Mash	**Looks** Add text strings, show and hide objects and change features.	**Looks** Set the appearance	**Looks** Set the appearance and create sequences of changes.	**Looks** Change the position of things between screen layers (send to back, bring to front)	**Looks** Change the position of things between screen layers (send to back, bring to front)
	Sound Select sounds, their duration and volume. Use programmes to create and record sound. Explore programmes to create different sounds and patterns. Purple Mash	**Sound** Select sounds, their duration and volume.	**Sound** Create and edit sounds	**Sound** Create and edit sounds - Control when they are heard, their volume, duration and rests	**Sound**	**Sound** Upload sounds and edit them - add effects such as fade in and out and control their implementation.
	Events Specify user inputs. To understand that pressing a control will have an effect – press go (Beebots) clicks to make things move.	**Events** Specify user inputs.	**Events** Specify condition to trigger events	**Events**	**Events** Set events to trigger events	**Events** Set events to trigger events 'broadcasting' information as a trigger

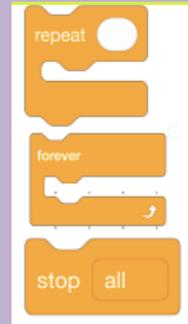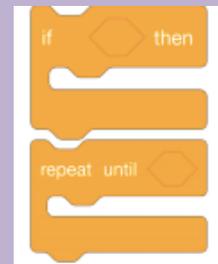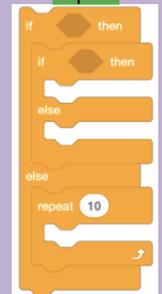| Control	Control	Control	Control	Control	Control
Specify the nature of events. To design a scene for a program, use code blocks to make the characters move automatically when the green Play button is clicked and to add an additional character who moves when clicked. Purple Mash	Specify the nature of events.	Repeat events.	Use IF THEN conditions to control events or objects.	Use IF THEN ELSE conditions to control events or objects	Use IF THEN ELSE conditions to control events or objects and repeat
Pen	**Pen**	**Pen**	**Pen**	**Pen**	**Pen**
Set pen colour, shape and size. To use programme Beebots to draw shapes. Purple Mash to draw programmed routes.	Set pen colour, shape and size.		Control the shades of pen.		Combine the use of pens with movement to create interesting effects.
Sensing	**Sensing**	**Sensing**	**Sensing**	**Sensing**	**Sensing**
Create conditions for actions. To explore the When Key and When Swiped commands (on tablets if available). • To use the Stop button to make characters stop when the background is clicked. Purple Mash and Beebots	Create conditions for actions.	Create conditions for actions by sensing proximity or by waiting for a user input (such as responses to questions).	Create conditions for actions by sensing proximity or by waiting for a user input (such as proximity to a specified colour or a line or responses to questions).	Use a range of sensing tools (including proximity, user inputs, loudness and mouse position) to control events or actions.	Use a range of sensing tools (including proximity, user inputs, loudness and mouse position) to control events or actions.
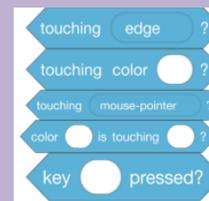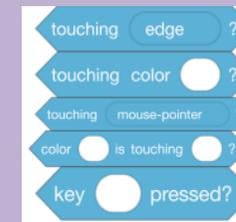
Operators	**Operators**	**Operators**	**Operators** Use reporter operators () + () () – () () * () () / () to perform calculations. 	**Operators** Use the Boolean operators i.e. () < () () = () () > () ()and() ()or() Not() to define conditions. 	**Operators** Use the Boolean operators i.e. () < () () = () () > () ()and() ()or() Not() to define conditions.
Variables	**Variables**	**Variables**	**Variables** Use variables and use the functions to control variables. Use the functions define, set, change, show and hide to control the variables. 	**Variables** Use lists to create a set of variables. 	
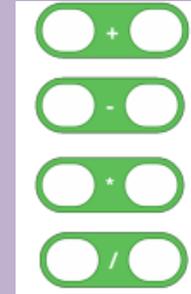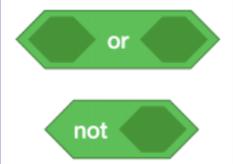
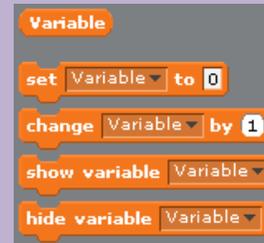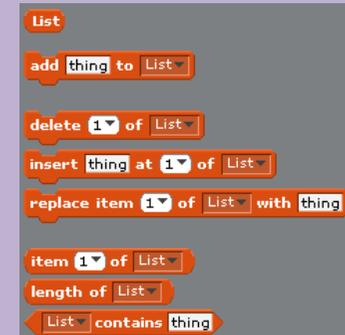|	Y1	Y2	Y3	Y4	Y5	Y6
To connect Digital Literacy: Children need to be able to use technology safely	Understand online risks. To join in sending a class email. To use a range of different technology and talk about its use. To follow the school's safer internet rules. To understand that some information is personal and should not be shared online.	Understand online risks. To send individual email in a controlled environment and reply. To use of a wide range of technology and can describe how it works in a variety of different contexts. To recognise that there are other people on the internet and this affects how they should use it. To use the internet safely for learning and communicating with others	Understand the term 'copyright.' Understand that comments made online that are hurtful or offensive are the same as bullying. Understand how online services work *Examples:* – *Understand that we can search for information in a variety of ways and that we influence the outputs of searches depending on our input* – *Know different ways of reporting unacceptable content and contact online* – *Understand when to share personal information and when not to* – *Understand that games and films have age ratings, and what that means* – *Understand that people can give permission for others to use their content*	Contribute to blogs, understand risks related to online bullying. Understand how online services work *Examples:* – *Recognise what kind of websites are trustworthy sources of information* – *Can rate a game or film they have made and explain their rating* – *Understand the benefits of a good password* – *Recognise the benefits and risks of different apps and websites* – *Understand that the media can portray groups of people differently*	Understand the effect of online comments and show responsibility and sensitivity when online. Understand how simple networks are set up and used. Understand and demonstrate knowledge that it is illegal to download copyrighted material, including music or games, without express written permission, from the copyright holder. *Examples:* – *Know where to find copyright free images and audio, and why this is important* – *Demonstrate responsible use of online services and technologies, and know a range of ways to report concerns* – *Evaluate websites for reliability of information and authenticity*	Understand the effect of online comments and show responsibility and sensitivity when online. Collaborate with others online, understand how networks are set up. Give examples of the risks of online communities and demonstrate knowledge of how to minimise risk and report problems.
To communicate Information Technology: Appropriate activities include word processing, creating images, taking and using photographs and video, creating music and animations, using and creating databases, producing websites and contributing to blogs. Most of it can be covered by using technology to support other subject areas though it may be necessary to teach some discrete skills.	Use a range of applications to communicate ideas, work and messages. To enter text using a keyboard. To record a sound and play it back. To use the space bar, backspace and return key. To create an image, add a title and text. Purple Mash, Microsoft Word.	Use a range of applications to communicate ideas, work and messages. To develop speed when typing and use a simple document with increasing control. To word process work, changing the font, font size, colour. To cut, copy and paste an image, text box, word art and clipart onto a document. To format text to refine and improve? e.g. underline, italics, bold. Purple Mash, Microsoft Word.	To use some advanced features of applications. *Key Skills:* *Open and save a file to a suitable folder* *Use suitable file names when saving work* *Use a search engine to find information using keyword searches* *Type using all fingers*	To use some advanced features of applications. *Key Skills:* *Use right-click, left-click and double-click appropriately on a mouse* *Use a search engine to find specific information* *Know how to copy text and images into another document*	Choose the most suitable application for purpose. *Key Skills:* *Use the keyboard confidently to type at a suitable pace* *Use common keyboard shortcuts* *Organise files effectively using folders*	Use many of the advanced features in order to create high quality, professional or efficient communications. *Key Skills:* *Use more advanced searching techniques when using a search engine* *Identify and use appropriate hardware and software to fulfil a specific task* *Evaluate their own content against success criteria and make improvements accordingly*

To collect			Devise and construct databases in areas across the curriculum.		Select appropriate applications to devise, construct and manipulate data and present it in an effective and professional manner.	
	Use simple databases. Examples: To contribute to adding information to a simple database and answer simple questions. To enter information into a template to make a graph or a pictogram. Purple Mash, J2data, Twinkle	Use simple databases. Examples: To enter information into a template to make a graph and pictogram. To present data in different ways. To use a branching database. Answer questions. Purple Mash, J2data, Twinkle	Examples: Explore a record database to find out information Know that there is a difference between data and information Use filters in a database to find out specific information Understand the benefits of using a computer to create charts and databases Understand that search engines store information in databases Design a questionnaire and collect a range of data on a theme	Examples: Enter data into a database and test Draw conclusions from information stored in a database, table or chart Understand that the Internet is made up of computers from all around the world connected together Understand that that school computers are connected together in a network Understand that we use a web browser to access information stored on the Internet Present data in a number of different ways to convey information	Examples: Appreciate that different programs work with different types of data, e.g. text, number Explore a record database to find out information Know that there is a difference between data and information Use filters in a database to find out specific information Understand the benefits of using a computer to create charts and databases	Examples: Understand that information can be stored and shared on the Internet Understand that search engines store information in databases Design a questionnaire and collect a range of data on a theme Enter data into a database package and test Draw conclusions from information stored in a database, table or chart

Every child has the right to an education. (Article 28)
Education must develop every child's personality, talents and abilities to the full. (Article 29)
Every child has the right to reliable information from a variety of sources. (Article 17)

Progression in Programming with Scratch

Planning Scratch Projects should be cross-curricular to:
- Meet the demands of a packed timetable
- Be more meaningful for the pupils
- Give pupils opportunity to apply the knowledge and understanding they have in other topic areas

However, please ensure that the application of pupils Computing skills are the primary focus.

Ideas for Scratch Projects

1. (Y3-4) Start by asking pupils to plan and then program a simple conversation between 2 sprites, for example:
    - English: dialogue between 2 main characters to represent a key part of a text
    - History: Dialogue between 2 main historical figures
    - MFL: Program a conversation in a different language (Scratch online has a built in translation tool or pupils could also use Google Translate to support)
    - PSHE: E-safety conversation
    - Geography: Pen Pal conversation about different parts of the world

    French Example (using 'Looks' blocks to share information and 'Motion' blocks to move the sprites)

    https://scratch.mit.edu/projects/55366092/editor/

2. Animations: Pupils can design and program an animation that demonstrates their learning within a topic, for example:
    - English: Animated version of a text being covered or animate the ending to a story, poem etc.
    - History: Facts about a particular topic (e.g. Ancient Greece).
    - PE: Use of 'Motion' and 'Say' blocks to demonstrate how to perform a particular skill
    - Geography: Rainforest animation

    History Example (incorporating 'Looks' blocks to share information, 'Sound' block, 'Event' blocks to change the background and 'Motion' blocks to move the sprite) https://scratch.mit.edu/projects/63956978/editor/

3. Quiz Game: Ask pupils to design and program a quiz game whereby the sprite responds according to the user's input

    Maths Quiz Example (Y4/5 objectives – including 'If' 'Then' Events blocks, setting a variable score, using 'Motion' blocks to ask questions, 'Operation' blocks) https://scratch.mit.edu/projects/97513873/editor/ (Next step - 8x table quiz incorporates more variables and sounds) https://scratch.mit.edu/projects/47561732/editor
    (Y5/6 objectives using lists to create a set of variables) https://scratch.mit.edu/projects/110408588/editor