

St Clements Primary School Progression Map

Subject: COMPUTING (COMPUTER SCIENCE)

At St Clements Primary School, we aim to prepare our learners for their future by giving them the opportunities to gain knowledge and develop skills that will equip them for an ever-changing digital world. Knowledge and understanding of computing is of increasing importance for children's future both at home and for employment. Our Computing curriculum focuses on a progression of skills in digital literacy, computer science, information technology and online safety to ensure that children become competent in safely using, as well as understanding, technology. These strands are revisited repeatedly through a range of themes during children's time in school to ensure the learning is embedded and skills are successfully developed. Our intention is that Computing also supports children's creativity and cross curricular learning to engage children and enrich their experiences in school.

Autumn	Nursery	EYFS	Key Stage 1		Key Stage 2			
		Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Y 6
Knowledge			<p>To understand that algorithms are precise instructions that can be followed.</p> <p>To follow a simple algorithm.</p> <p>To devise a simple algorithm.</p>	<p>To understand that an algorithm is a process that consists of a series of steps that achieve a specific goal.</p> <p>To understand that algorithms can describe everyday activities and can be followed by humans and computers.</p> <p>To understand the algorithms are made up of steps.</p>	<p>To program an animation that executes a sequence of statements.</p> <p>To understand that computer programs containing graphics use x and y coordinates and turns are measured in degrees.</p> <p>To understand that computer simulations can represent real or imaginary situations.</p> <p>To understand that computer simulations are guided by rules.</p> <p>To explore the effects of changing variables in a simulation using them to make and test predictions.</p>	<p>To understand that a program is a sequence of statements written in a programming language.</p> <p>To program a sequence of statements.</p>	<p>To understand that messages can be sent and received secretly</p> <p>To learn encrypt/decrypt simple messages.</p> <p>To understand that messages can be sent electronically over distances.</p> <p>To understand that computer programs containing graphics use x and y coordinates and turns are measure in degrees.</p> <p>To use conditional (if) statements.</p>	<p>To program a computer game by sequencing conditional statements.</p> <p>To use variable in programs.</p> <p>To use procedures in programs.</p>

Skills		<p>Follow and give simple instructions with help (algorithm)</p> <p>Make a programmable toy move but nit always as planned (programming)</p> <p>Use a limited set of software and tools to make something happen on screen but not always according to those planned.</p> <p>Identify simple repeating patterns.</p> <p>Sort a small set of objects according to criteria, sometimes with support</p> <p>Organize data into simple charts and graphs with support</p> <p>Answer questions using data with support.</p>	<p>Read a set of instructions and sometimes predict the correct outcome</p> <p>Produce instructions but sequence them incorrectly or make assumptions.</p> <p>Understand that humans and computers follow instructions.</p>	<p>Know that programming applications can be given commands to produce specific effects on screen.</p> <p>Produce a sequence of blocks that achieve a simple effect (e.g. move a sprite around the screen)</p> <p>Plan and give direct instructions to make things happen (e.g. playing robots)</p>	<p>Know that programming applications can be given commands to produce specific effects on screen.</p> <p>Produce a sequence of blocks that achieve a simple effect (e.g. move a sprite around the screen)</p> <p>Plan and give direct instructions to make things happen (e.g. playing robots)</p>	<p>Write an algorithm to produce a given effect using repetition. Accurately predict the outcome of a range of algorithms and programs.</p> <p>Test, debug and refine algorithms and programs</p> <p>Use sequence and basic selection and repetition in computer programs.</p> <p>Explain how a programmed effect has been achieved.</p> <p>Talk about improvements that could be made to progress.</p>	<p>Explain how a programmed effect has been achieved.</p> <p>Write and amend computer programs.</p> <p>Use repetition, variables, and conditional statements in computer programs.</p> <p>test, debug and refine algorithms and programs</p> <p>identify some common internet services that use the internet (e.g., online gaming or voice over internet)</p> <p>know that a computer takes input, processes it and produces output</p> <p>Identify a variety of computing devices and a number of inputs and outputs (e.g. touch, sound)</p> <p>know that computers store and manipulate data as a series of ones and zeros and that this is called binary</p>	<p>write or amend computer programs to produce specific actions</p> <p>understand that the same 'problem' can be solved in different ways</p> <p>know that commands can be given in shorter form</p> <p>use iteration (repeats and loops) in algorithms and programs</p>

Spring	EYFS	Key Stage 1			Key Stage 2		
	Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Knowledge		<p>To understand that programs execute by following precise and unambiguous instructions.</p> <p>To plan, test and debug a simple algorithm.</p> <p>To give instructions to programmable toy.</p> <p>To plan a simple algorithm to control a toy.</p>	<p>To know that the steps can be repeated.</p> <p>To understand that a computer need more precise instructions than a human.</p> <p>To use digital drawing tools (scratch) to create images.</p>	<p>To program a sequence of instructions that create visual effects.</p> <p>To import, create and record sounds.</p> <p>To understand that algorithms and programs can involve repetition.</p> <p>To understand that simulations can help people try things quickly and inexpensively.</p>	<p>To program an object to move and draw.</p> <p>To understand that commands and actions can be programmed to be executed depending upon whether a condition is true or not.</p>	<p>To understand that data can be transmitted as binary (on/off)</p> <p>To understand the algorithm of a simple shift cipher.</p> <p>To understand that some variables can only be true or false (Boolean)</p> <p>To understand that programs can do different things if the value of the Boolean variable is true or false (conditional statements)</p>	<p>To understand that the behaviour of a computer program should be planned.</p> <p>To understand that programs are developed according to a plan.</p>
Skills	<p>Give and follow simple instructions in order (algorithm)</p> <p>Create a short sequence of instructions (algorithms)</p> <p>Change instructions to create a different outcome (algorithm)</p> <p>Make a programmable toy move (programming)</p> <p>Use simple software and tools to make something happen</p> <p>Make choices on-screen about buttons and icons to select</p> <p>Create, recreate, and continue patterns</p> <p>Sort out a set of objects according to criteria</p> <p>Construct simple pictograms</p> <p>Know that pictures on a pictogram represent numerical values.</p>	<p>Read a set of instructions and usually predict the correct outcome.</p> <p>Produce a set of instructions that others can usually follow.</p> <p>Understand the computers follow instructions given in a precise way.</p>	<p>Produce a sequence of instructions that result in planned outcomes.</p> <p>Program a short sequence of commands that results in a planned effect.</p> <p>Program and test a simple program</p> <p>Create algorithms to solve simple problems.</p>	<p>Produce a sequence of instructions that result in planned outcomes.</p> <p>Program a short sequence of commands that results in a planned effect.</p> <p>Program and test a simple program</p> <p>Create algorithms to solve simple problems.</p>	<p>Write an algorithm to produce a given effect using repetition.</p> <p>Accurately predict the outcome of a range of algorithms and programs.</p> <p>Test, debug and refine algorithms and programs</p> <p>Use sequence and basic selection and repetition in computer programs.</p> <p>Explain how a programmed effect has been achieved.</p> <p>Talk about improvements that could be made to progress.</p>	<p>Write and amend more complex programs to create a variety of outcomes</p> <p>Program algorithms that achieve a range of specified outcomes</p> <p>create efficient programs by designing solutions using abstraction (E.g. using procedures in the form of 'My blocks' and/or broadcasts in Scratch)</p> <p>Test, debug and refine computer programs</p>	<p>Write and amend more complex computer programs to create a variety of outcomes</p> <p>decompose 'problems' by splitting them into smaller 'problems' and designing solutions for each part</p> <p>Use iteration(repeats and loops), variables and conditional statements (if..then) in computer programs</p> <p>Test computer programs and correct most errors</p>

Summer	EYFS	Key Stage 1		Key Stage 2			
	Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Knowledge		<p>To make predictions about an outcome based on a simple algorithm.</p> <p>To identify conditions and outcomes.</p> <p>To understand that some statements can only be true or false.</p> <p>To program a virtual object to move to on-screen objects.</p> <p>To record a sequence of instruction in a common format.</p>	<p>To program a simple animation involving movement.</p> <p>To write a simple program that produces an output (text)</p> <p>To combine images and text to create a simple animation.</p>	<p>To predict the outcome of a simple algorithm</p> <p>To combine images, sound, and movement to create a personal animation.</p> <p>To understand that simulations help us understand difficult concepts.</p> <p>To design and produce a computer simulation or adventure.</p>	<p>To combine repetition and conditional statements in a program.</p>	<p>To use frequency analysis to decipher encrypted text.</p> <p>To understand the importance of cryptography historically and today.</p> <p>To use variable in programs.</p>	<p>To develop strategies for testing and debugging computer programs.</p>
Skills	<p>Read a set of instructions and predict the outcome (algorithms)</p> <p>Write/draw a set of simple instructions in order (algorithms)</p> <p>Make changes to instruction and predict how the outcome will change (algorithms)</p> <p>Plan a set of instructions for a programmable toy and make it move (programming)</p> <p>Correct mistakes if instructions are incorrect (debugging)</p> <p>Talk about how their instructions could be improved</p> <p>Describe patterns and relationships</p> <p>Sort objects into sets according to one or more criteria</p> <p>Compare data using simple charts and graphs</p> <p>Suggest different ways data could be organized and displayed.</p> <p>Use graphs to answer a range of questions</p> <p>Create own questions that could be answered by interpreting data on a graph.</p> <p>Make comparisons between data on a graph.</p>	<p>Read a set of instructions and predict the correct outcome.</p> <p>Produce an accurate set of instructions using agreed language that others can follow.</p> <p>Understand that computers have no intelligence.</p>	<p>Design and develop basic computer programs.</p> <p>Combine sequences of commands into procedures (blocks of code_) that are repeated.</p> <p>Test and correct simple programs.</p> <p>Evaluate their own work and comment on improvements.</p>	<p>Design and develop basic computer programs.</p> <p>Combine sequences of commands into procedures (blocks of code_) that are repeated.</p> <p>Test and correct simple programs.</p> <p>Evaluate their own work and comment on improvements.</p>	<p>Explain how a programmed effect has been achieved.</p> <p>Write and amend computer programs.</p> <p>Use repetition, variables, and conditional statements in computer programs.</p> <p>test, debug and refine algorithms and programs</p> <p>identify some common internet services that use the internet (e.g., online gaming or voice over internet)</p> <p>know that a computer takes input, processes it and produces output</p> <p>Identify a variety of computing devices and a number of inputs and outputs (e.g. touch, sound)</p> <p>know that computers store and manipulate data as a series of ones and zeros and</p>	<p>Write and amend more complex programs to create a variety of outcomes</p> <p>Program algorithms that achieve a range of specified outcomes</p> <p>create efficient programs by designing solutions using abstraction (E.g. using procedures in the form of 'My blocks' and/or broadcasts in Scratch)</p> <p>Test, debug and refine computer programs</p>	<p>Create & use efficient methods of iteration, & nested conditional statements (if..then..else..if etc.)</p> <p>Systematically test computer programs for bugs and make them work as expected</p> <p>Critically analyse algorithms and programs and suggest more elegant solutions - e.g. by using abstraction to suggest single solution that could be used to solve a number of problems (i.e. procedures)</p> <p>Create procedures that call on other procedures (e.g. by using 'My blocks' and/or broadcasting blocks</p>

Impact (End Points)

EYFS	Key Stage 1		Key Stage 2			
Year R	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6