



Computing Skills Progression Grid

Early Years Outcome	KS1 National Curriculum Aims	KS2 National Curriculum Aims
<p>The main Early Years Outcomes covered in Computing are:</p> <ul style="list-style-type: none">• Children sing songs, make music and dance, and experiment with ways of changing them. They safely use and explore a variety of materials, tools and techniques, experimenting with colour, design, texture, form and function. (EAA-UMM ELG)• Children recognise that a range of technology is used in places such as homes and schools. They select and use technology for particular purposes. (UTT-T ELG)	<p>Pupils should be taught to:</p> <ul style="list-style-type: none">• 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• use technology purposefully to create, organise, store, manipulate and retrieve digital content recognise common uses of information technology beyond school• 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.	<p>Pupils should be taught to:</p> <ul style="list-style-type: none">• 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.
--	--	---

Hardware						
Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	<p>Learning how to explore and tinker with hardware to find out how it works</p> <p>Understanding that computers and devices around us use inputs and outputs, identifying</p>	<p>Understanding what a computer is and that it's made up of different components</p> <p>Recognising that buttons cause effects and that technology follows</p>	<p>Understanding what the different components of a computer do and how they work together</p> <p>Drawing comparisons across different types of computers</p>	<p>Learning about the purpose of routers</p>	<p>Learning that external devices can be programmed by a separate computer</p> <p>Learning the difference between ROM and RAM</p> <p>Recognising how the</p>	<p>Learning about the history of computers and how they have evolved over time</p> <p>Using the understanding of historic computers to design a computer of the</p>

	<p>some of these</p> <p>Learning where keys are located on the keyboard</p> <p>Learning how to operate a camera</p>	<p>instructions</p> <p>Learning how we know that technology is doing what we want it to do via its output.</p> <p>Using greater control when taking photos with tablets or computers</p> <p>Developing confidence with the keyboard and the basics of touch typing</p>	<p>Learning what a server does</p>		<p>size of RAM affects the processing of data</p> <p>Understanding the fetch, decode, execute cycle</p>	<p>future</p> <p>Learning how barcodes, QR codes and RFID work</p> <p>Learning about some of the methods which cause data corruption</p>
--	---	--	------------------------------------	--	---	--

Networks and Data

Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
			<p>Learning what a network is and its purpose</p> <p>Identifying the key components within a network, including whether they are wired or wireless</p>	<p>Consolidating understanding of the key components of a network</p> <p>Understanding that websites & videos are files that are shared from one computer to another</p>	<p>Learning the vocabulary associated with data: data and transmit</p> <p>Learning how the data for digital images can be compressed</p>	<p>Understanding that computer networks provide multiple services</p>

			<p>Recognising links between networks and the internet</p> <p>Learning how data is transferred</p>	<p>Learning about the role of packets</p> <p>Understanding that computer networks provide multiple services, such as the World Wide Web, and opportunities for communication and collaboration</p> <p>Develop strength, technique and flexibility throughout performances</p>	<p>Recognising that computers transfer data in binary and understanding simple binary addition</p> <p>Relating binary signals (Boolean) to the simple character-based language, ASCII</p> <p>Learning that messages can be sent by binary code, reading binary up to 8 characters and carrying out binary calculations</p> <p>Understanding how bit patterns represent images as pixels</p>	
Computational Thinking						
Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	<p>Learning that decomposition means breaking a problem down into smaller parts</p>	<p>Articulating what decomposition is</p> <p>Decomposing a game to predict</p>	<p>Using decomposition to explain the parts of a laptop computer</p> <p>Using decomposition</p>	<p>Solving unplugged problems by decomposing them into smaller parts</p>	<p>Decomposing animations into a series of images</p> <p>Decomposing a</p>	<p>Decomposing a program into an algorithm</p> <p>Using past</p>

	<p>Using decomposition to solve unplugged challenges</p> <p>Using logical reasoning to predict the behaviour of simple programs</p> <p>Developing the skills associated with sequencing in unplugged activities</p> <p>Learning that an algorithm is a set of step by step instructions used to carry out a task, in a specific order</p> <p>Follow a basic set of instructions</p> <p>Assembling instructions into a simple algorithm</p>	<p>the algorithms used to create it</p> <p>Using decomposition to decompose a story into smaller parts</p> <p>Learning what abstraction is</p> <p>Learning that there are different levels of abstraction</p> <p>Explaining what an algorithm is</p> <p>Following an algorithm</p> <p>Creating a clear and precise algorithm</p> <p>Learning that computers use algorithms to make predictions</p> <p>Learning that programs execute</p>	<p>to explore the code behind an animation</p> <p>Using repetition in programs</p> <p>Understanding that computers follow instructions</p> <p>Using an algorithm to explain the roles of different parts of a computer</p> <p>Using logical reasoning to explain how simple algorithms work</p> <p>Explaining the purpose of an algorithm</p> <p>Forming algorithms independently</p>	<p>Using decomposition to understand the purpose of a script of code</p> <p>Using decomposition to help solve problems</p> <p>Identifying patterns through unplugged activities</p> <p>Using past experiences to help solve new problems</p> <p>Using abstraction to identify the important parts when completing both plugged and unplugged activities</p> <p>Creating algorithms for a specific purpose</p>	<p>program without support</p> <p>Decomposing a story to be able to plan a program to tell a story</p> <p>Predicting how software will work based on previous experience</p> <p>Writing more complex algorithms for a purpose</p>	<p>experiences to help solve new problems</p> <p>Writing increasingly complex algorithms for a purpose</p>
--	--	--	---	---	---	--

		by following precise instructions				
		Incorporating loops within algorithms				
Programming						
Early Years	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
	<p>Programming a Bee-bot/Blue-bot to follow a planned route</p> <p>Learning to debug instructions when things go wrong</p> <p>Developing a how-to video to explain how the Bee-bot works. Learning to debug an algorithm in an unplugged scenario</p>	<p>Using logical thinking to explore software, predicting, testing and explaining what it does</p> <p>Using an algorithm to write a basic computer program</p> <p>Learning what loops are</p> <p>Incorporating loops to make code more efficient</p>	<p>Using logical thinking to explore more complex software; predicting, testing and explaining what it does</p> <p>Incorporating loops to make code more efficient</p> <p>Remixing existing code</p> <p>Using a more systematic approach to debugging code, justifying what is wrong and how it can be corrected</p>	<p>Understanding that websites can be altered by exploring the code beneath the site</p> <p>Coding a simple game</p> <p>Using abstraction and pattern recognition to modify code</p>	<p>Programming an animation</p> <p>Iterating and developing their programming as they work</p> <p>Beginning to use nested loops (loops within loops)</p> <p>Debugging their own code</p> <p>Writing code to create a desired effect</p> <p>Using a range of programming commands</p>	<p>Debugging quickly and effectively to make a program more efficient</p> <p>Remixing existing code to explore a problem</p> <p>Using and adapting nested loops</p> <p>Programming using the language Python</p> <p>Changing a program to personalise it</p> <p>Evaluating code to understand its purpose</p> <p>Predicting code and</p>

					Using repetition within a program	adapting it to a chosen purpose
					Amending code within a live scenario	Altering a website's code to create changes