

# Computer Science Progression of Skills



## Hardware

Nursery	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
<p>Operate a camera to take photographs.</p> <p>Identify familiar letters and numbers on a keyboard.</p> <p>Move the cursor on a mouse.</p>	<p>Learning how to operate a camera to take photographs of meaningful creations or moments.</p> <p>Learning how to explore and tinker with hardware to develop familiarity and introduce relevant vocabulary.</p> <p>Recognising and identifying familiar letters and numbers on a keyboard.</p> <p>Developing basic mouse skills such as moving and clicking.</p>	<p>Learning how to operate a camera or tablet to take photos and videos.</p> <p>Learning how to explore and tinker with hardware to find out how it works.</p> <p>Learning where keys are located on the keyboard.</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 instructions.</p> <p>Learning how we know that technology is doing what we want it to do via its output.</p> <p>Developing confidence with the keyboard and the basics of touch typing.</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>Using tablets or digital cameras to film a weather forecast.</p> <p>Understanding that weather stations use sensors to gather and record data which predicts the weather</p>	<p>Learning that external devices can be programmed by a separate computer.</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 future.</p> <p>Understanding and identifying barcodes, QR codes and RFID.</p> <p>Identifying devices and applications that can scan or read barcodes, QR codes and RFID.</p>

# Computer Science Progression of Skills

## *Networks and Data Representation*



Nursery	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
				<p>Understanding the role of the key components of a network.</p> <p>Identifying the key components within a network, including whether they are wired or wireless.</p> <p>Understanding that websites and videos are files that are shared from one computer to another.</p> <p>Learning about the role of packets.</p> <p>Understanding how networks work and their purpose.</p> <p>Recognising links between networks and the internet.</p> <p>Learning how data is transferred.</p>	<p>Understanding that computer networks provide multiple services, such as the World Wide Web, and opportunities for communication and collaboration.</p>	<p>Learning the vocabulary associated with data: data and transmit.</p> <p>Recognising that computers transfer data in binary and understanding simple binary addition.</p> <p>Learning that messages can be sent by binary code, reading binary up to eight characters and carrying out binary calculations.</p>	

# Computer Science Progression of Skills



## *Computational Thinking*

Nursery	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Predict what will happen if following simple instructions given.	Using logical reasoning to understand simple instructions and predict the outcome.	<p>Learning that decomposition means breaking a problem down into smaller parts.</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>Following a basic set of instructions.</p> <p>Assembling instructions into a simple algorithm.</p>	<p>Articulating what decomposition is. Decomposing a game to predict the algorithms used to create it.</p> <p>Learning that there are different levels of abstraction.</p> <p>Explaining what an algorithm is. Following an algorithm.</p> <p>Creating a clear and precise algorithm.</p>	<p>Using decomposition to explain the parts of a laptop computer.</p> <p>Using decomposition to explore the code behind an animation.</p> <p>Using repetition in programs.</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 solve a problem by finding out what code was used.</p> <p>Using decomposition to understand the purpose of a script of code.</p> <p>Identifying patterns through unplugged activities.</p> <p>Using abstraction to identify the important parts when completing both plugged and unplugged activities.</p>	<p>Decomposing animations into a series of images.</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>Decomposing a program into an algorithm.</p> <p>Using past experiences to help solve new problems.</p> <p>Writing increasingly complex algorithms for a purpose.</p>

# Computer Science Progression of Skills



## Programming

Nursery	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
<p>Following instructions as part of practical activities and games.</p> <p>Give two steps or instructions.</p>	<p>Following instructions as part of practical activities and games.</p> <p>Learning to give simple instructions.</p> <p>Learning to debug instructions, with the help of an adult, when things go wrong.</p>	<p>Programming a Floor robot to follow a planned route.</p> <p>Learning to debug instructions when things go wrong.</p> <p>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>Using logical thinking to explore more complex software; predicting, testing and explaining what it does.</p> <p>Incorporating loops to make code more efficient. Continuing existing code.</p>	<p>Creating algorithms for a specific purpose. Coding a simple game.</p> <p>Using abstraction and pattern recognition to modify code.</p> <p>Incorporating variables to make code more efficient.</p>	<p>Iterating and developing their programming as they work.</p> <p>Confidently using loops in their programming.</p> <p>Using a more systematic approach to debugging code, justifying what is wrong and how it can be corrected.</p> <p>Writing code to create a desired effect.</p> <p>Using a range of programming commands.</p> <p>Using repetition within a program.</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. Predicting code and adapting it to a chosen purpose.</p>