

Computing Progression

	Progression of Content					
	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Algorithms	<p>Begin to understand what an algorithm is. Begin to write a simple set of instructions for a purpose using symbols and check for errors (debug). Begin to predict the behaviour of simple programs.</p>	<p>Knows what an algorithm is and is able to express simple algorithms as symbols and can check for errors (debug) and correct. Understands that computers need precise instructions. Understands that algorithms are used on digital devices as programs. Designs simple algorithms using loops, and selection (as statements). Demonstrates logical reasoning to predict outcomes.</p>	<p>Creates algorithms that use repetition and two-way selection (i.e. if, then, else). Uses diagrams to express solutions. Uses logical reasoning to predict outputs, showing an awareness of inputs.</p>	<p>Understands and explains which tasks are best completed by humans or computers. Designs solutions by decomposing a problem and creates a sub-solution for each part of the problem (decomposition). Understands that there are several solutions to the same problem.</p>	<p>Can explain which tasks are best completed by humans or computers. Designs solutions by decomposing a problem and creates sub-solutions for each part of the problem (decomposition). Understands that there are several solutions to the same problem. Knows that various algorithms exist for different functions. Begins to identify patterns in algorithms that help to solve specific problems.</p>	<p>Knows that iteration is the repetition of a process, such as a loop. Understands that different algorithms exist for the same problem. Finds errors in algorithms, re-writes and tests own sequences. Can identify similarities and differences in situations and can use these to solve problems.</p>
Communication and networks	<p>Obtains content from the world wide web using a web browser. Understands the importance of communicating safely and respectfully online, and the need to keep personal information private.</p>	<p>Uses the web and can carry out simple web searches. Uses computers safely and responsibly, knowing a range of ways to report unacceptable content and contact when online. Understands the need for adult supervision.</p>	<p>Understands the difference between the internet and an internet service e.g. WWW. Shows an awareness of, and can use internet services such as VOIP. Recognises what is acceptable and unacceptable behaviour when using technologies and online services</p>	<p>Knows how to effectively use search engines, and knows how search results are selected. Chooses, combines and uses internet services. Uses technology and online services responsibly and knows a range of ways to report concerns.</p>	<p>Understands how search engines rank search results, and tests some of these systems. Knows how to construct static web pages using HTML and CSS. Understands data transmission between digital computers over networks, including the internet i.e. IP addresses and packet switching.</p>	<p>Knows how search engines rank search results. Can construct static web pages using HTML and CSS. Designs and creates own web pages for a purpose. Knows about data transmission between digital computers over networks, including the internet i.e. IP addresses and packet switching.</p>

Computing Progression

Data and data representation	Knows that digital content can be represented in many forms. Begin to distinguish between some of these forms and can explain the different ways that they communicate information. Sorts, stores, edits and manipulates data in different digital formats.	Create, organise, store, manipulate and retrieve digital content.	Can explain the difference between data and information. Knows why sorting data in a file can improve searching for information. Uses filters and can perform single criteria searches for information.	Carries out more complex searches for information. Analyses and evaluates data and information, and recognises that poor quality data leads to unreliable results and inaccurate conclusions	Understands the function of the main internal parts of basic computer design (architecture). Starts to show understanding of the concepts behind the fetch- execute cycle. Starts to understand that there is a range of operating systems and application software for the same hardware.	Can explain that digital computers use binary to represent all data. Knows that bit patterns represent numbers and images. Knows that computers transfer data in binary (code). Can explain the relationship between binary and file size (uncompressed) Defines data types: real numbers and Boolean. Queries data on one table using a typical query language.
Hardware and processing	Understands that computers have no intelligence and can do nothing unless a program is used. Understands that all software used on digital devices is programmed.	Knows that a range of digital devices can be considered a computer (can name examples). Can identify and uses a range of input and output devices (e.g. robotics). Understands how programs specify the function of a general purpose computer.	Understands that computers collect data from various input devices e.g. sensors and application software. Can explain the difference between hardware and software and their roles within a computer system	Knows why and when computers are used. Knows the main functions of the operating system. Knows the difference between physical, wireless and mobile networks. Can give examples e.g. internet; how they provide multiple services such as the world-wide web.	Understands how search engines rank search results, and tests some of these systems. Knows how to construct static web pages using HTML and CSS. Understands data transmission between digital computers over networks, including the internet i.e. IP addresses and packet switching.	Understands the function of the main internal parts of basic computer design. Knows the concepts behind the fetch-execute cycle. Knows that there is a range of operating systems and application software for the same hardware. Can test, contrast and evaluate the effectiveness
Information technology	Uses software under supervision, to create, store and edit digital content using appropriate file and folder names. Understands that people interact with computers.	Uses technology independently, to organise digital content. Shows awareness for the quality of digital content collected. Uses software to manipulate and present	Confidently collects, organises and presents data and information in digital content. Produces digital content to achieve a given goal through combining software packages and	Makes judgements about digital content when evaluating and assigning it for a given audience. Considers the audience when designing and creating digital content.	Evaluates the appropriateness of digital devices, internet services and application software to achieve given goals. Appreciates ethical issues surrounding the	Combines and uses multiple digital devices, internet services and application software to achieve given goals and explains choices.

Computing Progression

	Explains their use of technology in school. Understands how IT is used outside school. Talks about their work and makes changes to improve it.	digital content, data and information. Talks about their experience of technology in school and outside school. Discusses their work and makes improvements to solutions based on feedback received.	internet services to communicate with a wider audience e.g. blogging. Makes effective improvements to solutions based on feedback received, and evaluates success of the solution.	Appreciates the potential of IT for collaboration when computers are networked. Applies criteria to evaluate the quality of solutions. Identifies improvements, making refinements to the solution, and future solutions.	application of IT beyond school. Designs criteria to critically evaluate the quality of solutions. Uses the criteria to identify improvements and can make appropriate refinements to the solution.	Evaluates the trustworthiness of digital content. Understands how the use of technology can impact on society. Has some knowledge of design criteria for users to evaluate the quality of solutions and uses the feedback to identify some improvements.
Programming and development	Understands that users can develop their own programs. Shows understanding by creating simple programs e.g. programmable robots. Executes, checks and changes programs. Knows that programs work by following precise instructions.	Creates their own programs e.g. robots. Uses arithmetic operators, what if statements and loops within programs. Uses logical reasoning to predict the behaviour of programs. Finds and corrects simple semantic errors (debugging). Recognises the different types of data e.g. text and number. Appreciates that programs can work with different types of data. Understands that data can be presented in tables to make it useful. Confidently organises, stores, edits and manipulates data in a range of digital formats. Begins to understand the difference between data and information.	Designs programs that implement algorithms to achieve given goals. Finds and assigns variables in programs. Uses loop commands until and sequences of selection statements in programs, including if, then, else statements.	Knows the differences between and appropriately uses if , then and else statements. Uses variable and relational operators within a loop to control programs. Designs, writes and debugs programs using procedures (algorithms). Understands that a procedure can be used to hide details in programs. Understands, and can clearly explain, the difference between data and information. Knows why sorting data in a file can improve searching for information	Knows that programming bridges the gap between algorithmic solutions and computers. Has practical experience of high-level textual language, e.g. standard libraries when programming. Uses some operators and expressions e.g. Boolean. Begins to apply them in the context of program control (e.g. input / process / output).	Knows that programming bridges the gap between algorithmic solutions and computers. Has practical experience of high-level textual language, e.g. standard libraries when programming. Uses a range of operators and expressions e.g. Boolean and applies them in the context of program control. Chooses the appropriate data types