

KS1 Progression

	Computer Science			Information Technology	Digital Literacy	
NC Objectives	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.
Year 1	Understand that an algorithm is a set of instructions used to solve a problem or achieve an objective. Know that an algorithm written for a computer is called a program.	Work out what is wrong with a simple algorithm when the steps are out of order and write own simple algorithm. Know that an unexpected outcome is due to a code that has been created and make logical attempts to fix the code.	Look at a program and read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. Predict/interpret where the turtle in challenges will end up at the end of the program.	Sort, collate, edit and store simple digital content e.g. name, save and retrieve work and follow simple instructions to access online resources.	Understand what is meant by technology and identify a variety of examples both in and out of school. Make a distinction between objects that use modern technology and those that do not.	Understand the importance of keeping information, such as usernames and passwords, private and actively demonstrate this in lessons. Take ownership of work and save work in own designated space.
Year 2	Explain that an algorithm is a set of instructions to complete a task. When designing simple programs, show an awareness of the need to be precise with algorithms so that they can be successfully converted into code.	Create a simple program that achieves a specific purpose and also identify and correct some errors. Program designs display a growing awareness of the need for logical, programmable steps.	Identify the parts of a program that respond to specific events and initiate specific actions. For example, write a cause and effect sentence of what will happen in a program.	Demonstrate an ability to organise data using, for example, a database and retrieve specific data for conducting simple searches. Edit more complex digital data such as music compositions. With confidence; create, name, save and retrieve content. Use a range of media when creating digital content including photos, text and sound.	Effectively retrieve relevant, purposeful digital content using a search engine. Apply learning of effective searching beyond the classroom. Make links between; technology they see around them, coding and multimedia work they do in school e.g. animations, interactive code and program	Know the implications of inappropriate online searches. Begin to understand how things are shared electronically. Develop an understanding of using email safely and know ways of reporting inappropriate behaviours and content to a trusted adult.

KS2 Progression

	Computer Science				Information Technology		Digital Literacy
NC Objectives	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 concern about content and contact.
Year 3	Turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. Design shows the thought processes of the desired task and how this translates into code. Identify errors within programs (that prevents it following the desired algorithm) and then fix it.	Demonstrate the ability to design and code a program that follows a simple sequence. Experiment with timers to achieve repetition effects in programs. Begin to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects. Understand how variables can be used to store information while a program is executing.	Designs for programs show the thinking of the structure of a program in logical, achievable steps and absorbing some new knowledge of coding structures. For example, 'if' statements, repetition and variables. Make good attempts to 'step through' more complex code in order to identify errors in algorithms and can correct this. In programs such as Logo, 'read' programs with several steps and predict the outcome accurately.	List a range of ways that the internet can be used to provide different methods of communication. Use some of these methods of communication, e.g. being able to open, respond to and attach files to email. Describe appropriate email conventions when communicating in this way.	Carry out simple searches to retrieve digital content and understand that to do this, they are connecting to the internet and using a search engine such as internet-wide search engines.	Collect, analyse, evaluate and present data and information using a selection of software, e.g. using a branching database or using software. Consider what software is most appropriate for a given task. Create purposeful content to attach to emails.	Demonstrate the importance of having a secure password and not sharing this with anyone else. Explain the negative implications of failure to keep passwords safe and secure. Understand the importance of staying safe and the importance of one's conduct when using familiar communication platforms. Know more than one way to report unacceptable content and contact.
Year 4	When turning a real-life situation into an algorithm, designs show the thought processes of the desired task and how this translates into code using coding structures for selection and repetition. Make more intuitive attempts to debug programs that they have created.	Use of timers to achieve repetition effects are more logical and are integrated into their program designs. Understand 'if statements' for selection and attempt to combine these with other coding structures including variables to achieve the effects that they design in programs. Understand how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables. Make use of user inputs and outputs.	Designs for programs show that the structure of a program has been thought out in logical, achievable steps and there is opportunities to absorb some new knowledge of coding structures. For example, 'if' statements, repetition and variables. Trace code and use step-through methods to identify errors in code and make logical attempts to correct this. In programs such as Logo, 'read' programs with several steps and predict the outcome accurately.	Recognise the main component parts of hardware which allow computers to join and form a network. Understanding of the online safety implications, associated with the ways the internet can be used to provide different methods of communication is improving.	Understand the function, features and layout of a search engine. Appraise selected webpages for credibility and information at a basic level	Make improvements to digital solutions based on feedback. Make informed software choices when presenting information and data. Create linked content using a range of software Share digital content within their community, i.e. using Virtual Display Boards.	Explore key concepts relating to online safety using concept mapping. Help others to understand the importance of online safety. Know a range of ways of reporting inappropriate content and contact.

KS2 Progression

	Computer Science				Information Technology		Digital Literacy
NC Objectives	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 concern about content and contact.
Year 5	Attempt to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. Test and debug programs as they go and use logical methods to identify the approximate cause of any bug, but may need some support identifying the specific line of code.	Translate algorithms that include sequence, selection and repetition into code with increasing ease. Own designs show the thought processes to accomplish the set task in code, utilising such structures. Combine sequence, selection and repetition with other coding structures to achieve intended algorithm design.	When coding, opportunities are provided to think about one's code structure in terms of the ability to debug and interpret the code later, e.g. the use of tabs to organise code and the naming of variables	Understand the value of computer networks but are also aware of the main dangers. Recognise what personal information is and can explain how this can be kept safe. Select the most appropriate form of online communications contingent on audience and digital content.	Carry out searches with greater complexity for digital content when using a search engine. Able to explain in some detail how credible a webpage is and the information it contains.	Able to make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the solution. Objectively review solutions from others. Collaboratively create content and solutions using digital features within software. Able to use several ways of sharing digital content.	Develop secure knowledge of common online safety rules and apply this by demonstrating the safe and respectful use of a few different technologies and online services. Implicitly relate appropriate online behaviour to their right to personal privacy and mental wellbeing of themselves and others.
Year 6	Able to turn a more complex programming task into an algorithm by identifying the important aspects of the task (abstraction) and then decomposing them in a logical way, using prior knowledge of possible coding structures and applying skills from previous programs. Test and debug programs as they go and use logical methods to identify the cause of bugs, demonstrating a systematic approach to try to identify a particular line of code causing a problem.	Translate algorithms that include sequence, selection and repetition into code. Own designs show the thought processes to accomplish the set task in code, utilising such structures, including nesting structures within each other. Coding displays an improving understanding of; variables in coding, outputs such as sound and movement, inputs from the user of the program such as, button clicks and the value of functions.	Ability to interpret a program in parts and can make logical attempts to put the separate parts of a complex algorithm together to explain the program as a whole.	Understand and explain in some depth, the difference between the internet and the World Wide Web. Know what a WAN and LAN are and can describe how they access the internet in school.	Readily apply filters when searching for digital content. Able to explain in detail how credible a webpage is and the information it contains. Compare a range of digital content sources and rate them in terms of content quality and accuracy. Use critical thinking skills in everyday use of online communication.	Make clear connections to the audience when designing and creating digital content. Design and create own blogs to become a content creator on the internet, Able to use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.	Demonstrate the safe and respectful use of a range of different technologies and online services. Identify more discreet inappropriate behaviours through developing critical thinking. Recognise the value in preserving one's privacy when online for one's own and other people's safety