Early years	Digital literacy		Information technology	Computer science
National curriculum Ks1	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	Recognise common uses of information technology beyond school	Use technology purposefully to create, organise, store, manipulate and retrieve digital content	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
Skills (to show progression)				Remember what an algorithm is. Explore using algorithms Interpret simple algorithms Alter algorithms
Resources/ Ideas	E-safety: childnet resources			Beebots - Physical - Ipad app - Create and program sprites using scratch junior
Glossary	Instruction-algorithm Control Digital content - as text, images, sound, video, animation, virtual environments, multimedia Information Internet Program			Search Choose/select a program Order Software World wide web

Year 1	Digital literacy		Information technology	Computer science		
National curriculum Ks1	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	Recognise common uses of information technology beyond school	Use technology purposefully to create, organise, store, manipulate and retrieve digital content	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
Skills	Understand the importance of keeping information, such as their usernames and passwords, private and actively demonstrate. Take ownership of their work and save this in their own private space.	Understand what is meant by technology and can 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 e.g. a microwave vs. a chair.	Sort, collate, edit and store simple digital content e.g. Children can name, save and retrieve their work and follow simple instructions to access online resources. (sorting shapes), (manipulating backgrounds) or using pictogram software	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 Write their own simple algorithm Know that an unexpected outcome is due to the code they have created make logical attempts to fix the code	Read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. (predict what will happen)

Resources/ Ideas	Save to my work folder on purple mash or google drive	Sorting tech vs non tech Learn vocabulary	Log in Save to my work folder on purple mash or google drive 2quiz example (sorting Shapes), 2code design mode 2code design mode (manipulating backgrounds) Or using pictogram 2count.	Https://www.scratchjr.org/teach/activities Introduction 1. Drive across the city 2. Run a race 3. Sunset 4. Moonrise after sunset 5. Spooky forest

Year 2	Digital literacy		Information technology	Computer science		
National curriculum Ks1	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	Recognise common uses of information technology beyond school	Use technology purposefully to create, organise, store, manipulate and retrieve digital content	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
Skills	Know the implications of inappropriate online searches. Begin to understand how things are shared electronically they develop an understanding of using email Know ways of reporting inappropriate behaviours and content to a trusted adult.	Effectively retrieve relevant, purposeful digital content using a search engine. Apply their learning of effective searching beyond the classroom. Share this knowledge make links between technology they see around them, coding and multimedia work they do in school	Demonstrate an ability to organise data using, for example, a database and can retrieve specific data for conducting simple searches. Edit more complex digital data such as music compositions Confident when creating, naming, saving and retrieving content. Use a range of media in their digital content including photos, text and sound.	Children can explain that an algorithm is a set of instructions to complete a task. When designing simple programs, children show an awareness of the need to be precise with their algorithms so that they can be successfully converted into code.	Children can create a simple program that achieves a specific purpose. Identify and correct some errors. Children's program designs display a growing awareness of the need for logical, programmable steps.	Children can identify the parts of a program that respond to specific events and initiate specific actions. For example, they can write a cause and effect sentence of what will happen in a program.
Resources/ Ideas	Research Google classroom comments Email safely by using 2respond activities on purple mash	Google classroom Share this knowledge, e.g. 2publish example template. E.g. Animations, interactive code and programs.	Stop frame animations Music w tom A database such as 2investigate	Https://www.scratchjr.org/teach/activities 5. Spooky forest 6. Dribble a basketball 7. Dance party 8. Meet and greet. 9. Conversation Designing simple programs,		

Year 3	Digital literacy	Information technology	Computer science

National curriculum Ks2	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact	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	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
Skills	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 their conduct when using familiar communication tools. Know more than one way to report unacceptable content and contact.	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 search engines.	Collect, analyse, evaluate and present data and information using a selection of software. Consider what software is most appropriate for a given task. They can create purposeful content to attach to emails.	Turn a simple real-life situation into an algorithm for a program by deconstructing it into manageable parts. Design shows that they are thinking of the desired task and how this translates into code. Identify an error within their program 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 their programs. Beginning 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 their programs showing that they are 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. 'Read' programs with several steps and predict the outcome accurately. (In programs such as logo)
Resources/ Ideas	2email in purple mash. E-safety lesson plans. Be internet legends Digital citizenship game: interland be internet awesome	Email using 2email Purplemash email conversation with santa. Chatbot/ email back and forth ahmed's secret.	To use search engine kidrex	Using a branching database (2question), using software such as 2graph. Attach to emails using 2respond.	Scratch Https://codeclubprojects.org/en-gb/scratch/ Module 1 1. Rockband 2. Lost in space 3. Ghost busters		

Year 4	Digital literacy		Information technology		Computer science		
National curriculum Ks2	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact	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	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
Skills	Explore key concepts relating to online safety Help others to understand the importance of online safety. Know a range of ways of reporting inappropriate content and contact.	Recognise the main component parts of hardware which allow computers to join and form a network. Their ability to understand the online safety implications associated with the ways the internet can be used to provide different	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. They create linked content using a range of software Share digital content within their community.	When turning a real life situation into an algorithm, the children's design shows that they are thinking of the required task and how to accomplish this in code using coding structures for selection and repetition.	Use of timers to achieve repetition effects are becoming 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 their programs. Understand how variables can be used to store information	Children's designs for their programs show that they are 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. Trace code and use step-through methods to identify errors in code and make logical attempts to correct this.

		methods of communication is improving.			Make more intuitive attempts to debug their own programs.	while a program is executing, they are able to use and manipulate the value of variables. Make use of user inputs and outputs	'read' programs with several steps and predict the outcome accurately. (in programs such as logo).
Resources/ Ideas	Online safety using concept mapping such as 2connect. E-safety-be internet legends			Range of software such as 2connect and 2publish+. Share digital content, i.e. Using virtual display boards.	Https://codeclubprojects.org/en-gb/scratch/ Module 1 4. Chatbot 5. Paint box 6. Boat race		

Year 5	Digital literacy		Information technology		Computer science		
National curriculum Ks2	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact	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.	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.
Skills	Have a secure knowledge of common online safety rules and can 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.	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.	Search with greater complexity for digital content when using a search engine. Explain in some detail how credible a webpage is and the information it contains.	Make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the solution. E.g. Creating their own program to meet a design brief. Objectively review solutions from others. Collaboratively create content and solutions using digital features within software such as collaborative mode. They are able to use several ways of sharing digital content.	Attempt to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. Test and debug their programs as they go and can 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 and their own designs show that they are thinking of how to accomplish the set task in code utilising such structures. They are combining sequence, selection and repetition with other coding structures to achieve their algorithm design.	When children code, they are beginning to think about their 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.
Resources/ Ideas	E-safety :be internet legends	E.g. 2blog, 2email, display boards.	Boolean operators Https://www.youtube.com/watch?V=rl3jmegmcse	Own program to meet a design brief using 2code Sharing digital content, i.e. 2blog, display boards and 2email.	Scratch Https://codeclubprojects.org/en-gb/scratch/ Module 2 1. Memory 2. Dodgeball 3. Brain game		

Year 6	Digital literacy		Information technology		Computer science		
National curriculum Ks2	Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact	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.	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.
Skills	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 their privacy when online for their own and other people's safety.	Understand and can 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.	Apply filters when searching for digital content. Explain in detail how credible a webpage is and the information it contains. Compare a range of digital content sources and are able to 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 their own blogs to become a content creator on the internet Use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.	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 their knowledge of possible coding structures and applying skills from previous programs. Test and debug their program 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 and their own designs show that they are thinking of how 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.	Children are able 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.
Resources/ Ideas	Critical thinking, e.g. 2respond activities.			E.g. 2blog.	Scratch Https://codeclubprojects.org/en-gb/scratch/ Module 2 4. Catch the dots 5. Clone wars 6. Create your own world		

Glossary

Algorithm – an unambiguous procedure or precise step-by-step guide to solve a problem or achieve a particular objective.

Computer networks – the computers and the connecting hardware (wifi access points, cables, fibres, switches and routers) that make it possible to transfer data using an agreed method ('protocol').

Control – using computers to move or otherwise change 'physical' systems. The computer can be hidden inside the system or connected to it.

Data – a structured set of numbers, representing digitised text, images, sound or video, which can be processed or transmitted by a computer.

Debug – to detect and correct the errors in a computer program.

Digital content – any media created, edited or viewed on a computer, such as text (including the hypertext of a web page), images, sound, video (including animation), or virtual environments, and combinations of these (i.e. Multimedia).

Information – the meaning or interpretation given to a set of data by its users, or which results from data being processed.

Input – data provided to a computer system, such as via a keyboard, mouse, microphone, camera or physical sensors.

Internet – the global collection of computer networks and their connections, all using shared protocols (tcp/ip) to communicate.

Logical reasoning – a systematic approach to solving problems or deducing information using a set of universally applicable and totally reliable rules.

Output – the information produced by a computer system for its user, typically on a screen, through speakers or on a printer, but possibly though the control of motors in physical systems.

Program – a stored set of instructions encoded in a language understood by the computer that does some form of computation, processing input and/or stored data to generate output.

Repetition – a programming construct in which one or more instructions are repeated, perhaps a certain number of times, until a condition is satisfied or until the program is stopped.

Search – to identify data that satisfies one or more conditions, such as web pages containing supplied keywords, or files on a computer with certain properties.

Selection – a programming construct in which the instructions that are executed are determined by whether a particular condition is met.

Sequence – to place programming instructions in order, with each executed one after the other.

Services – programs running on computers, typically those connected to the internet, which provide functionality in response to requests; for example, to transmit a web page, deliver an email or allow a text, voice or video conversation.

Simulation – using a computer to model the state and behaviour of real-world (or imaginary) systems, including physical and social systems; an integral part of most computer games.

Software – computer programs, including both application software (such as office programs, web browsers, media editors and games) and the computer operating system. The term also applies to ‘apps’ running on mobile devices and to web-based services.

Variables – a way in which computer programs can store, retrieve or change simple data, such as a score, the time left, or the user’s name.

World wide web – a service provided by computers connected to the internet (web servers), in which pages of hypertext (web pages) are transmitted to users; the pages typically include links to other web pages and may be generated by programs automatically

Scratch progression - https://projects.raspberrypi.org/en/codeclub

Module 1

Module 2