Progression of Knowledge in Computing

Over a year curriculum time in Computing should have the following weighting 50% Computer Science, 25% Digital Literacy and 25% Information Technology

	KS1		Year 3 and 4		Year 5 and 6	
	Year A	**Year B**	**Year A**	**Year B**	**Year A**	**Year B**
	Substantive Knowledge					
Computer Science (Coding, Programming)	Know that an algorithm is a set of instructions used to solve a problem or achieve an outcome. Know that an algorithm written for a computer is called a program.	Know and explain that an algorithm is a set of clearly sequenced instructions to complete a task Know that each step in the instructions needs to be precise in an algorithm	Know how to turn a simple real-life situation into an algorithm for a program Create a program design that can be translated into code	Know how to turn a real life situation into an algorithm, using coding structures for selection and repetition Use their knowledge of coding	Know how to turn more complex real-life situations into algorithms for a program by deconstructing it into manageable parts. Know how to test and debug	Know how 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 their knowledge of

	Know how to work out what is wrong with a simple algorithm when the steps are out of order When looking at a program, children know to read code one line at a time	Know how to identify and correct some errors in a program they have written Write a cause and effect sentence of what will happen in a program. Know how to use the repeat command in an algorithm Know how to use numerical commands in a simple algorithm	Know how to identify an error within their program that prevents it following the desired algorithm and then fix it. Know how to use timers to achieve repetition effects in their programs Understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects	structures when debugging an algorithm Know how to use timers to achieve logical repetition effects and how to integrate them into their program designs. Understand 'if statements' for selection Know how to combine coding structures including if statements and variables to achieve the effects that	their programs as they go Know how to identify the specific line of code to debug Know how to translate algorithms that include sequence, selection and repetition into code Know how to combine sequence, selection and repetition with other coding structures to achieve their coding design Know how to use tabs to organise	possible coding structures and applying skills from previous programs. Know how to test and debug their program as they go and use logical methods to identify the cause of bugs Know how to create nesting structures as part of their code. Know how to translate algorithms that include sequence, selection and repetition into code utilising coding structures,

			Know how to use if statements when creating a simple program Know and understand how variables can change the outcome of a program Identify errors in algorithms with more complex code and can correct the errors. Know that the internet can be used to provide different methods of communication Use Google Classroom as a	they design in their programs. Know how to use and manipulate the value of variables. Know and can explain the function of user inputs and outputs Use simple inputs and outputs to create a desired outcome Know how to trace code and use step-through methods to identify	code and name variables within a program or series of algorithms Know and understand the value of computer networks Know and can identify the main dangers of computer networks Know and can explain what personal information is and how this can be kept safe when communicating online	including nesting structures within each other. Know and can use outputs such as sound and movement. Know how to incorporate these outputs into their coding designs. Know and can use inputs from the user such as button clicks as part of their coding designs. Know and understand the use of variables, outputs and inputs in coding. Know and understand the value of functions

			means to communicate Know how to share, respond to and turn in files Know how to create a new file	errors in code. Know the main component parts of hardware which allow computers to join and form a network Make links between the different methods of communicatio n the internet can offer and the online safety implications	Know how to select the most appropriate form of online communications based on both the intended audience and digital content	within a coding design or series of algorithms Know how to interpret a program in parts Know how to put the separate parts of a complex algorithm together to explain the program as a whole, in a logical way. Know what a WAN and LAN are and can describe how they access the internet in school
	Disciplinary Knowledge					
	Write their own simple algorithm	Show an awareness of the	Know how to deconstruct an	Create coding designs that	Use logical methods to	Demonstrate a systematic

	Know that an unexpected outcome is due to the code they have created Know how to make logical attempts to fix the code Make some attempts to envision the bigger picture of the overall effect of the program Predict the outcome of the program	need to be precise with their algorithms so that they can be successfully converted into code. Create a simple program that achieves a specific purpose Know how to identify the parts of a program that respond to specific events and initiate specific actions	algorithm into manageable parts Design a program that shows an understanding of the desired outcome Design and code a program that follows a simple sequence Know and understand how variables can be used to store information while a program is executing. Create programs that show that they are thinking of the structure of a	show that they are thinking of the required task and how to accomplish this in code Be more intuitive in attempts to debug their own programs. Understand how variables can be used to store information while a program is executing Be able to identify errors in code and make logical	identify the approximate cause of any bug in a program Create coding designs that show that they are thinking of how to accomplish the set task in code utilising the coding structures of sequence, selection, and repetition Consider the code structure in terms of the ability to debug and interpret the code later	approach to try to identify a particular line of code causing a problem. Create coding designs that show that they are thinking of how to accomplish the set task in code using a wide range of structures including sequencing, selection, repetition, variables and nesting structures Know, understand and can explain in some depth the difference between the

			program in logical, achievable steps and absorbing some new knowledge of coding structures Know how to read programs with several steps and predict the outcome accurately.	attempts to correct this In coding programs read programs with several steps and predict the outcome accurately.		internet and the World Wide Web
Information Technology	**Substantive Knowledge**					
	Know how to sort digital content Know how to collate and combine information Know how to edit their work	Know how to organise information into a database Know how to search a database for specific information	Know how to carry out simple searches to retrieve digital content using a search engine Know how to collect, analyse, evaluate and	Know and understand the function, features and layout of a search engine Know how to ask specific questions	Know how to search with greater complexity for digital content when using a search engine, using inverted commas and and/or search terms.	Know how to apply filters when searching for digital content Compare a range of digital content sources and are able to rate them in terms of

	Know how to save and retrieve their work Know how to take a photograph and focus using the zoom functions Know how to add a filter to a photograph Know how to animate a picture by recording sound Know and use the sound recording icons	Know how to use a simple search engine Know how to combine sounds into a simple sequence Know how to to edit more complex digital data such as music compositions Know how to create, name, save and retrieve digital content Know how to animate an image or character	present data and information Know how to create a branching database Know how to create a range of different graphs and use them to ask and answer questions Know how to combine text and images in a presentation or document Know how to add sound and video to a presentation Know how to add transitions to a	when using a search engine Know how to create short video clips to use as part of a multimedia presentation Know how to create and combine text, images, video and sound when producing a multimedia presentation Know how to present information using branching databases, graphing programmes	Know and explain in some detail how credible a webpage is and the information it contains. Know how to create their own program to meet a design brief Know and use several ways of sharing digital content with others	content quality and accuracy Know how to design and create content that is hosted on the internet such as blogs, podcasts and webpages. Know how to identify improvements to their own digital content, making some refinements.

			presentation to create a specific effect	and simple spreadsheet. Know how to link digital content using hyperlinks		
	Disciplinary Knowledge					
	Follow simple instructions to access online resources	Use a range of media in their digital content including photos, text and sound.	Understand that when using a search engine they are connecting to the internet Consider what software is most appropriate for a given task and give reasons for the choice Create purposeful content in response to a given task	Use the different features and functions of a search engine Begin to appraise selected webpages for credibility and information Share digital content within their community Make improvements	Understand and explain the criteria for evaluating a web-page for credibility and plausibility. Make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the solution Objectively review solutions	Explain in detail how credible a web-page is and the information it contains Use critical thinking skills in everyday use of online communication. Make clear connections to the audience when designing and creating digital content Use criteria to evaluate the

				to digital solutions based on feedback. Make informed software choices when presenting information and data	from others and give an informed opinion with reasons. Collaboratively create content and solutions using digital features within software	quality of digital solutions
Digital Literacy	**Substantive Knowledge**					
	Know and understand what is meant by technology and can identify a variety of examples both in and out of school. Know and can describe objects that use modern technology and those that do not	Know how to retrieve relevant, purposeful digital content using a search engine Know the implications of inappropriate online searches Understand how things are shared electronically	Know and can explain the negative implications of failure to keep passwords safe and secure. Know and understand the importance of staying safe and the importance of their conduct when using familiar	Know and can explain appropriate content, conduct and contact online Know a range of ways of reporting inappropriate content and contact.	Know and can explain a range of common online safety rules and can apply this knowledge by demonstrating the safe and respectful use of a few different technologies and online services	Know and can identify more discreet inappropriate behaviours through developing critical thinking. Know and recognise the value in preserving their privacy when online for their

	Know and can talk about ways to stay safe online	Know ways of reporting inappropriate behaviours and content to a trusted adult.	communication tools such as Google Classroom Know more than one way to report unacceptable content and contact.			own and other people's safety.
	Disciplinary Knowledge					
	Understand the importance of keeping information, such as their usernames and passwords, private Take ownership of their work and save this in their own private space such as Google Classroom, Class Dojo portfolio or Seesaw	Make links between technology they see around them, coding and multimedia work they do in school	Demonstrate the importance of having a secure password and not sharing this with anyone else	Children can explore key concepts relating to online safety using concept mapping They can help others to understand the importance of online safety.	Relate appropriate online behaviours to their right to personal privacy and mental wellbeing of themselves and others.	Demonstrate the safe and respectful use of a range of different technologies and online services.