Progression in Computing

Intent:

At Ralph Butterfield Primary School, we understand the immense value that technology plays not only in supporting the Computing and whole school curriculum but overall in the day-to-day life of our school. Our aims are to fulfil the requirements of the National Curriculum for Computing whilst also providing enhanced collaborative learning opportunities, engagement in rich content and supporting pupil's conceptual understanding of new concepts, which support the needs of all our pupils.

"A high-quality computing education equips pupils to use computational thinking and creativity to understand and change the world…core of computing is computer science, in which pupils are taught the principles of information and computation, how digital systems work, and how to put this knowledge to use through programming. Building on this knowledge and understanding, pupils are equipped to use information technology to create programs, systems and a range of content." National Curriculum

Our Computing curriculum aims to develop the heart and mind of every child. Computing teaching at Ralph Butterfield Primary School has deep links with mathematics, science and design and technology and our aim is to provide a broad and balanced curriculum whilst ensuring that pupils become digitally literate and digitally resilient. Technology is ever evolving and we aim to develop pupils who can use and express themselves, develop their ideas through, information and communication technology at a suitable level for the future workplace and as active participants in a digital world.

The aims of our Computing curriculum are to develop pupils who:

- Are responsible, competent, confident and creative users of information and communication technology.
- Know how to keep themselves safe whilst using technology and on the internet and be able to minimise risk to themselves and others.
- Become responsible, respectful and competent users of data, information and communication technology.
- Can evaluate and apply information technology, including new or unfamiliar technologies, analytically to solve problems.
- Can analyse problems in computational terms, and have repeated practical experience writing computer programs in order to solve such problems.
- Can understand and apply the fundamental principles and concepts of computer science, including abstraction, logic, algorithms and data representation.
- Become digitally literate and are active participants in a digital world.
- Are equipped with the capability to use technology throughout their lives.
- Understand the importance of governance and legislation regarding how information is used, stored, created, retrieved, shared and manipulated.
- Have a 'can do' attitude when engaging with technology and its associated resources.
- Utilise computational thinking beyond the Computing curriculum.
- Understand and follow the SMART E-Safety rules.
- Understand the E-Safety messages can keep them safe online.
- Know who to contact if they have concerns.
- Apply their learning in a range of contexts, e.g. at school and at home.
- Know where to locate the CEOP button and how to use it.

	Reception	Year 1	Year 2	Year 3	Year 4	Year 5	Year 6
Programme of Study	No computing in Statutory Framework for Early Years or ELG **Birth to Five Matters:** Children require access to a range of technologies, both digital and non-digital in their early lives. Exploring with different technologies through play provides opportunities to develop skills that children will go on to develop in their lifetimes. Investigations, scientific inquiry and exploration are essential components of learning about and with technology both digitally and in the natural world. Through technology children have additional opportunities to learn across all areas in both formal and informal ways. Technologies should be seen as tools to learn both from and with, in order to integrate technology effectively within early years practice	Pupils should be taught to: •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.		Pupils should be taught to: •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 concerns about content and contact.			
Computer Science		Children understand that an algorithm is a set of instructions used to	Children can explain that an algorithm is a set of instructions to	Children can turn a simple real-life situation into an algorithm for a	When turning a real-life situation into an algorithm, the children's	Children may attempt to turn more complex real-life situations into	Children are able to turn a more complex programming task

		solve a problem or achieve an objective. They know that an algorithm written for a computer is called a program.	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.	program by deconstructing it into manageable parts. Their design shows that they are thinking of the desired task and how this translates into code. Children can identify an error within their program that prevents it following the desired algorithm and then fix it.	design shows that they are thinking of the required task and how to accomplish this in code using coding structures for selection and repetition. Children make more intuitive attempts to debug their own programs.	algorithms for a program by deconstructing it into manageable parts. Children are able to 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.	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. Children 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.
		Children can work out what is wrong with a simple algorithm when the steps are out of order, e.g. The Wrong Sandwich in Purple Mash and can write their own simple algorithm, e.g. Colouring in a Bird activity.	Children can create a simple program that achieves a specific purpose. They can also identify and correct some errors, e.g. Debug Challenges: Chimp. Children's program designs display a	Children demonstrate the ability to design and code a program that follows a simple sequence. They experiment with timers to achieve repetition effects in their programs.	Children's use of timers to achieve repetition effects are becoming more logical and are integrated into their program designs. They understand 'if statements' for selection and attempt to	Children can 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	Children 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,

		Children know that an unexpected outcome is due to the code they have created and can make logical attempts to fix the code, e.g. Bubbles activity in 2Code.	growing awareness of the need for logical, programmable steps.	Children are beginning to understand the difference in the effect of using a timer command rather than a repeat command when creating repetition effects. Children understand how variables can be used to store information while a program is executing.	combine these with other coding structures including variables to achieve the effects that they design in their programs. As well as understanding how variables can be used to store information while a program is executing, they are able to use and manipulate the value of variables. Children can make use of user inputs and outputs such as 'print to screen'. e.g. 2Code.	utilising such structures. They are combining sequence, selection and repetition with other coding structures to achieve their algorithm design.	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.
		When looking at a program, children can read code one line at a time and make good attempts to envision the bigger picture of the overall effect of the program. Children can, for example, interpret where the turtle in 2Go challenges will end up at the end of the program.	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.	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. They make good attempts to 'step	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. They can trace code and use step-through	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.	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.

				through' more complex code in order to identify errors in algorithms and can correct this. e.g. traffic light algorithm in 2Code. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.	methods to identify errors in code and make logical attempts to correct this. e.g. traffic light algorithm in 2Code. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately.		
				Children can list a range of ways that the internet can be used to provide different methods of communication. They can use some of these methods of communication, e.g. being able to open, respond to and attach files to emails using 2Email. They can describe appropriate email conventions when communicating in this way	Children 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 methods of communication is improving.	Children understand the value of computer networks but are also aware of the main dangers. They recognise what personal information is and can explain how this can be kept safe. Children can select the most appropriate form of online communications contingent on audience and digital content, e.g. 2Blog, 2Email, Display Boards.	Children understand and can explain in some depth the difference between the internet and the World Wide Web. Children know what a WAN and LAN are and can describe how they access the internet in school.

Information Technology	Children begin to log on to a computer (Summer Term) Children begin to double click on an icon to know that it gets them to the program they want to use.	Children are able to 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, use Purple Mash 2Quiz example (sorting shapes), 2Code design mode (manipulating backgrounds) or using pictogram software such as 2Count.	Children demonstrate an ability to organise data using, for example, a database such as 2Investigate and can retrieve specific data for conducting simple searches. Children are able to edit more complex digital data such as music compositions within 2Sequence. Children are confident when creating, naming, saving and retrieving content. Children use a range of media in their digital content including photos, text and sound.	Children can carry out simple searches to retrieve digital content. They understand that to do this, they are connecting to the internet and using a search engine such as Purple Mash search or internet-wide search engines.	Children understand the function, features and layout of a search engine. They can appraise selected webpages for credibility and information at a basic level.	Children search with greater complexity for digital content when using a search engine. They are able to explain in some detail how credible a webpage is and the information it contains.	Children readily apply filters when searching for digital content. They are able to explain in detail how credible a webpage is and the information it contains. They compare a range of digital content sources and are able to rate them in terms of content quality and accuracy. Children use critical thinking skills in everyday use of online communication.
				Children can collect, analyse, evaluate and present data and information using a selection of software, e.g. using a branching database	Children are able to make improvements to digital solutions based on feedback. Children make informed software choices when	Children are able to make appropriate improvements to digital solutions based on feedback received and can confidently comment on the success of the	Children make clear connections to the audience when designing and creating digital content. The children design and create their own blogs to become a

				(2Question), using software such as 2Graph. Children can consider what software is most appropriate for a given task. They can create purposeful content to attach to emails, e.g. 2Respond.	presenting information and data. They create linked content using a range of software such as 2Connect and 2Publish+. Children share digital content within their community, i.e. using Virtual Display Boards.	solution. e.g. creating their own program to meet a design brief using 2Code. They objectively review solutions from others. Children are able to 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, i.e. 2Blog, Display Boards and 2Email.	content creator on the internet, e.g. 2Blog. They are able to use criteria to evaluate the quality of digital solutions and are able to identify improvements, making some refinements.
Digital Literacy		Children understand what is meant by technology and can identify a variety of examples both in and out of school. They can make a distinction between objects that use modern technology and those that do not e.g. a microwave vs. a chair.	Children can effectively retrieve relevant, purposeful digital content using a search engine. They can apply their learning of effective searching beyond the classroom. They can share this knowledge, e.g. 2Publish example template.	Children demonstrate the importance of having a secure password and not sharing this with anyone else. Furthermore, children can explain the negative implications of failure to keep passwords safe and secure. They understand the	Children can explore key concepts relating to online safety using concept mapping such as 2Connect. They can help others to understand the importance of online safety. Children know a range of ways of reporting	Children 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. Children implicitly relate appropriate online behaviour to	Children demonstrate the safe and respectful use of a range of different technologies and online services. They identify more discreet inappropriate behaviours through developing critical thinking, e.g. 2Respond activities. They

			Children make links between technology they see around them, coding and multimedia work they do in school e.g. animations, interactive code and programs.	importance of staying safe and the importance of their conduct when using familiar communication tools such as 2Email in Purple Mash. They know more than one way to report unacceptable content and contact.	inappropriate content and contact.	their right to personal privacy and mental wellbeing of themselves and others.	recognise the value in preserving their privacy when online for their own and other people's safety.
		Children understand the importance of keeping information, such as their usernames and passwords, private and actively demonstrate this in lessons. Children take ownership of their work and save this in their own private space such as their My Work folder on Purple Mash	Children know the implications of inappropriate online searches. Children begin to understand how things are shared electronically such as posting work to the Purple Mash display board. They develop an understanding of using email safely by using 2Respond activities on Purple Mash and know ways of reporting inappropriate behaviours and content to a trusted adult.				

End Points Computer Science	• To know that by following the instructions correctly, they will get the correct result. • To know how to make a beebot move forwards and backwards and make a turn left and right.	• To know how to follow and create simple instructions on the computer. • To know that by following the instructions correctly, they will get the correct result. • To know how to make a character move left and right. • To know that an algorithm is a precise, step-by-step set of instructions used to solve a problem or achieve an objective. • To know that computers need precise instructions to follow. • To know that an algorithm written for a computer to follow is called a program. • To know how the order of instructions affects the result. • To know that correcting errors in an algorithm or	• To know that an algorithm is a set of instructions. • To know how to use commands. • To know what debugging means. • To know how to use algorithms successfully to achieve an end result. • To know how to sort information using a binary tree.	• To know how to design and write a program that accomplishes a specific goal. • To know how to design and write a program that simulates a physical system. • To know how to use repetition commands. • To begin to understand 'if' statements. • To begin to understand how the use of the timer differs from the repeat command and can experiment with the different methods of repeating blocks of code. • To know what steps to follow to debug a program. • To begin to understand variables. • To understand what a variable is in programming.	• To know what Object, Action, Output, Control and Event are in computer programming. • To know which commands they included in their program and what they achieve. • To know how to use 'If/else' statements. • To understand what a variable is in programming. • To know what steps I need to follow to debug a program. • To know what they did so that their computer program would not work. • To know what a variable is when used in programming. • To know how they made their program change the number every second. • To know what the different instructions are in Logo and how to type them.	• To know some ways that text variables can be used in coding.	• To know how to design and write a more complex program that accomplishes a specific goal. • To know what functions are and how they can be created and labelled. • To know how buttons are used in work. • To know how to code a text adventure.

		program is called 'debugging'. •To understand the functionality of the basic direction keys. •To understand how to create and debug a set of instructions (algorithm). •To understand how to change and extend the algorithm list. •To know what is meant by coding. •			•To know some of the language of Logo. •To know how to use Logo to write letters. •To know how to use the build feature in Logo.		
END POINT Information Technology	•To understand that data can be represented in picture format. •To know how to use a pictogram to record the results of a question •To know the difference between a traditional book and an e-book.	•To know how to sort items using a range of criteria. •To understand that data can be represented in picture format. •To know how to sort items using the 'Grouping' activities. •To know how to use a pictogram to record the results of an experiment. •To know the difference between a traditional book and an e-book.	•To know how to use a spreadsheet to add amounts. •To know what is meant by a binary tree. •To know what is meant by a database. •To know how to use a paint program to create art based upon different artistic styles. •To know how to make music digitally •To know how a story can be	•To know how to create pie charts and bar graphs. •To know how to use the 'more than', 'less than' and 'equals' tools. •To know the names of the fingers. •To know what is meant by 'top row', 'home row', 'bottom row' and 'space bar'. •To begin to know typing terminology. •To know how to send an email.	•To know how to use the formula wizard and add formulae and explore formatting cells. •To know how to use data to create a line graph. •To know how to use a spreadsheet for budgeting •To know how to use place value in spreadsheets. •To know how to write for different purposes. •To know if a piece of writing is	•To know how to use the 'how many' tool. •To know how to convert measurements in spreadsheets. •To know how to create a formulae including the advanced mode. •To know how to use text variables to perform calculations. •To know how to use a spreadsheet to plan an event. •To know the different ways to search a database.	•To know how to create and use spreadsheets. •To know the purpose, audience and features of writing blogs. •To know that blogs need to be updated regularly to maintain the audience's interest and engagement. •To know how to create quizzes, considering the audience and the type of questions that are best suited.

		• To know how to add animation to a story. • To know how to add sound to a story including voice recording and music the children have created. • To know how to use additional features to enhance their stories. • To know how to add images to a spreadsheet and use the image toolbox. • To know how to navigate around a spreadsheet. • To know how to use the 'speak' and 'count' tools to count items.	presented in different ways. • To know that digital content can be represented in many forms. • To know that data can be structured in tables to make it useful. • To know how to use a variety of software to manipulate and present digital content and information.	• To understand how to complete a branching database. • To know how to use and debug their own branching database. • To know that a computer simulation can represent real and imaginary situations. • To know how to analyse and evaluate a simulation. • To know how to enter data into a graph and answer questions. • To know how to present the results in graphic form.	suitable for its audience. • To know what makes a good animated film or cartoon. • To know how animations are created by hand. • To know what the onion skinning tool does in animation. • To know what stop motion animation is and how it is created. • To know how to locate information on the search results page.	• To know how to create a database around a chosen topic. • To know what a database field is and correctly add field information. • To know how to create the game environment. • To begin to understand the process of designing their own game. • To know how to finish and share their game. • To know what the 2Design and Make tool is for. • To understand designing for a purpose. • To understand printing and making. • To begin to understand the possibilities of 3D printing. • To understand the need for visual representation when generating and discussing complex ideas.	

						• To know the importance of recording concept maps visually. • To know how to use the correct vocabulary when creating a concept map. • To know how a concept map can be used to retell stories and information.	
END POINT Digital Literacy	• To know how to search Purple Mash to find resources. • To know what is meant by 'technology. • To know types of technology used in school and out of school.	• To know how to log in safely. • To understand the idea of 'ownership' over their creative work. • To know how to find saved work in the Online Work area and find teacher comments. • To know how to search Purple Mash to find resources. • To know how to use the different icons to add pictures and text to their work. • To understand the importance of	• To know how to refine searches using the search tool. • To know how to share work electronically using the display boards. • To know how things can be shared electronically for others to see both on Purple Mash and the Internet. • To know how to communicate with others if they are not in front of you. • To know the meaning of key internet terms.	• To understand what makes a good password for use on the Internet. • To know that some information held on websites may not be accurate or true. • To know how to send an email safely.	• To know about online safety. • To know the name of the different parts of a desk top. • To know the function of the different parts of a computer. • To know whether an information source is true and reliable.	• To know what internet safety is. • To know the importance of keeping personal information safe. To understand issues concerning the reliability of sources and people online. • To know what Childnet SMART CREW is and have used their resources to gain an understanding about keeping safe online. • To know who to tell if I they upset by something that happens online.	• To have a good understanding of the various areas of online safety that they have studied throughout school. • To know the safety aspects of blogging. • Demonstrate an awareness of the issues surrounding inappropriate posts and cyberbullying. • To know the difference between the World Wide Web and the Internet.

		logging out when they have finished. • To know about common icons used, e.g. Save, Print, Open, New. • To know what is meant by 'technology. • To know types of technology used in school and out of school.					