Computing: Year Group: 3

Prior Learning Year 2	Year 3 Learning	Year 4 Future Learning	Vocabulary - Subject Specific	Linked Vocabulary
Computer Science: 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. They can also 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. **Information Technology** Pupils will learn to recognise that different devices can be used to capture photographs and will gain experience capturing, editing, and improving photos. Finally, they will use this knowledge to recognise that images they see may not be real. Pupils are able to sort, collate, edit and store simple digital content e.g. children can name, save and retrieve their work using the Purple	**Computer Science:** Children can turn a simple real-life situation into an algorithm for a 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. 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 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. 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 Pupils compare digital and non-digital devices, before being introduced to computer networks that include network infrastructure devices like routers and switches. **Information Technology** Pupils will carry out simple searches to retrieve digital content. They understand that to do this, they are connecting to	**Computer Science:** When turning a real-life situation into an algorithm, the pupil'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. Children make more intuitive attempts to debug their own programs. Pupils 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 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. Pupils can trace code and use step-through methods to identify errors in code and make logical attempts to correct this. In programs such as Logo, they can 'read' programs with several steps and predict the outcome accurately. Pupils will apply their knowledge and understanding of networks, to appreciate the internet as a network of networks which need to be kept secure. They will learn that the World Wide Web is part of the internet, and be given opportunities to explore the	Posture, keys, typing, appropriate, blog, inappropriate, password, personal information, internet, reputable source, permission, spoof, reliable source, verify, vlog, website, action, alert, algorithm, background, bug, button, click event, code, collision detection event, command, debug/ debugging, event, flowchart, implement, input, interval, nesting, object, predict, properties, repeat, scene, sequence, test, timer, turtle object, Binary tree, branching database, database, data, debugging, Axis, chart, column, data, graph, investigation, row, sorting, tally chart	Posture, keys, Appropriate, inappropriate, personal information, permission, reliable, verify, action, alert, background, bug, button code, collision, event, flowchart, implement, input, interval, nesting, object, predict, properties, repeat, scene, sequence, test, timer, object, database, data, axis, chart, column, data, graph, investigation, row, sorting, tally chart

Mash application '2Count'. **Digital Literacy:** Pupils explore how IT benefits society in places such as shops, libraries, and hospitals. Whilst discussing the responsible use of technology, and how to make smart choices when using it.	the internet and using a search engine such as Purple Mash search or internet-wide search engines. Pupils will use slide show software to create a presentation. They will learn how to add pages, include media, customize animations and add timings. Pupils will create their own 'branching database' and be able to sort objects using 'yes' or 'no' questions. **Digital Literacy:** Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact See 'Online Safety'.	World Wide Web for themselves to learn about who owns content and what they can access, add, and create. Finally, they will evaluate online content to decide how honest, accurate, or reliable it is, and understand the consequences of false information. **Information Technology** Pupils will understand the function, features and layout of a search engine. They can appraise selected webpages for credibility and information at a basic level. Pupils are able to make improvements to digital solutions based on feedback. Pupils make informed software choice when presenting information and data. Pupils create linked content using a range of software such as 2Connect and 2Publish+. Pupils share digital content within their community e.g. using virtual display boards. **Digital Literacy:** Use technology safely, respectfully and responsibly; recognise acceptable/unacceptable behaviour; identify a range of ways to report concerns about content and contact See 'Online Safety'.		

Common Misconceptions	Key Questions:	Famous People Links	
• Misunderstanding of key vocabulary. • Effective typing. • Good password practice. • WWW is complete factual. • Age limits for games and website.	• Why should I have a good posture at the computer? • Why should I type certain keys with certain fingers? • What is a password and why should we keep them safe? • Is everything I read on the internet true? • How do I know if I am old enough to play a computer game?	• Charles Babbage – First person to make a mechanical computer. • Alan Turing – Mathematician who famously helped break Germany's Enigma code by design a computer to decipher the code. • John Von Neumann – Mathematician who developed computer architecture. E.g. memory (RAM).	

• Uses for spreadsheet and data collection.	• Why is it useful to use a flowchart to design a computer program? • What does repeat mean in computer programming? • What is the difference between 'timer after' and 'timer every'? • What is meant by data? • What is a database? • What is a branching database? • What is a graph? • What are the frame lines on the graph called? • What different kinds of graphs are there?	• Douglas Engelbart – pioneer in the development of modern computers. • Steve Jobs – Co-founder of Apple which invented iPad, iPhone, Apple Mac. • Philip Don Estridge – Developed the first IBM personal computer which paved the way for universal parts/ peripherals. • Bill Gates – Founder of Microsoft. • Tim Berners-Lee – invented the WWW. • https://www.sutori.com/en/story/famous-people-in-computer-history--TcHp7hWrDd1ZfLW2zQfxCs5h

Assessment Opportunities/Final Assessment

- FFT – Termly Assessments
- Continuous assessment (AfL / formative).