



Brookland Federation - Brookland Junior School				
Progression of Skills - Computing				
	3	4	5	6
Computer Science	<p>Understand that a micro:bit is programmable hardware with inputs and outputs.</p> <p>Identify and recognise key hardware components of the micro:bit.</p> <p>Use block-based code to create simple programs.</p> <p>Create programs that use inputs to trigger outputs (e.g. LEDs, text, sound).</p> <p>Use event-driven commands to control program behaviour.</p> <p>Understand that programs follow a sequence and that the order of commands matters.</p> <p>Use repetition (forever loops) and timing (sleep) to control program flow and create effects.</p> <p>Test and refine programs using a simulator before running them on physical hardware.</p>	<p>Use directional and control commands to program movement (e.g. Logo turtle).</p> <p>Create programs that use sequence and repetition to achieve a goal, including regular shapes.</p> <p>Write, save and improve procedures, considering efficiency.</p> <p>Debug programs logically, including correcting errors in commands and spacing.</p> <p>Understand that inputs can be detected by sensors (e.g. light, accelerometer).</p> <p>Write programs that respond to sensor inputs.</p> <p>Understand that variables store values and can change during program execution.</p> <p>Use variables with selection (if / else) to control outcomes.</p> <p>Recognise and use control structures (sequence, repetition, selection).</p> <p>Understand how inputs, processing (code) and outputs work together to create simple systems.</p>	<p>Design, plan and create programs that meet a specific purpose, using decomposition to break problems into parts and abstraction to remove unnecessary detail.</p> <p>Develop algorithms for real-world systems and simulations and convert them into programs that use sequence, selection and repetition, including timed loops.</p> <p>Create programs with multiple objects, using shared variables or properties and functions (procedures) to improve efficiency and organisation.</p> <p>Use numeric and string variables that change in response to events or time and display their values as outputs.</p> <p>Test, evaluate and refine programs, comparing them to real-life behaviour where appropriate and using feedback to improve functionality, playability and user experience.</p>	<p>Understand and use different representations of data, including denary and binary, and convert simple numbers between the two to explain how computers store information.</p> <p>Create programs that use inputs, variables and selection (if / else) to control outcomes, including timed sequences and on/off states.</p> <p>Use coordinates (x and y) to control object position and interaction within programs.</p> <p>Organise and manage larger programs using tabs and functions (procedures) to improve structure, readability and debugging.</p> <p>Create and reuse multiple functions to produce more complex outcomes than in previous years.</p> <p>Interpret flowcharts to understand, predict and test how programs will run, linking algorithms to code.</p> <p>Select and use appropriate input methods (e.g. prompts or input commands) within programs.</p> <p>Test, debug and refine programs to ensure they run as intended.</p>
Information Technology	<p>Use digital tools to communicate appropriately, including email, and understand basic online safety risks.</p> <p>Create and edit simple presentations,</p>	<p>Plan and create digital animations, using storyboards to sequence ideas.</p> <p>Understand how technology improves efficiency compared to</p>	<p>Create, organise and interrogate databases to answer questions using searches and queries.</p> <p>Create and format documents that</p>	<p>Use advanced spreadsheet features to analyse, sort and present data effectively.</p> <p>Create and refine formulae and</p>



	<p>combining text, images and media for a purpose.</p> <p>Enter, edit and interpret data in simple spreadsheets, using charts and basic formulae.</p> <p>Use spreadsheets to support real-life tasks, such as simple budgeting.</p>	<p>non-digital methods.</p>	<p>combine text, images, tables and links for a specific audience and purpose.</p> <p>Use spreadsheets and formulae to model situations, analyse data and support decision-making.</p>	<p>charts that update automatically when data changes.</p> <p>Use spreadsheets to model scenarios and plan events, understanding the benefits of automation.</p>
<p>Digital Literacy</p>	<p>Understand different communication methods (digital and non-digital) and choose appropriately.</p> <p>Use email to communicate within school.</p> <p>Recognise key features of an email interface.</p> <p>Send, receive and respond to emails, including attachments.</p> <p>Identify basic risks when communicating digitally (e.g. attachments).</p>	<p>Use search engines effectively with appropriate keywords.</p> <p>Understand how search engines find, rank and present information.</p> <p>Refine searches and evaluate the usefulness of results.</p> <p>Distinguish between fact, opinion and belief.</p> <p>Identify unreliable or fake news and explain why it exists.</p> <p>Check the reliability of online information.</p> <p>Understand what artificial intelligence is and describe its impact, including benefits and concerns.</p>	<p>Understand and manipulate 3D digital models.</p> <p>Design and refine a digital product to meet a brief.</p> <p>Use digital tools to edit, improve and finalise work.</p> <p>Understand file formats and the process from digital design to physical outcome (e.g. printing).</p> <p>Evaluate and refine work before publication or production.</p>	<p>Plan, create and publish digital content for an audience.</p> <p>Structure content for clarity, engagement and purpose.</p> <p>Use drafting, editing and reviewing processes.</p> <p>Interact responsibly with online audiences through comments.</p> <p>Apply netiquette and recognise inappropriate online behaviour.</p> <p>Know how to report content that breaches community or legal guidelines.</p>