

Year 2 Computing Curriculum



Autumn	Computing systems and networks: What is a computer?	Programming: Algorithms and Debugging
Year 2	- The difference between a desktop and a laptop computer. - People control technology. - Some input devices that give a computer an instruction about what to do (output). - Computers often work together.	- To understand what machine learning is and how it enables computers to make predictions. - To know that loops in programming are where you set a certain instruction (or instructions) to be repeated multiple times. - To know that abstraction is the removing of unnecessary detail to help solve a problem.
Spring	Computing systems and networks: Word Processing	Code.org
Year 2	- Touch typing is the fastest way to type. - I can make text a different style, size and colour. - Copy and paste is a quick way of duplicating text.	- Define a list of steps (algorithm) to get a friend from their starting position to their goal - Identify and fix errors in the execution of an algorithm - Translate a list of steps into a series of physical actions - Experiment with standard block-based programming actions such as: clicking, drag and drop, etc. - Build a computer program from a set of written instructions. - Construct a program by reorganising sequential movements. - Translate movements into a series of commands. - Determine a sequence of movements necessary to solve a puzzle. - Identify and locate bugs in a program. - Translate movements into a series of commands.
Summer	Data handling: International Space Station	Code.org
Year 2	- Simple data can be entered into a spreadsheet. - What steps are needed to take to create an algorithm. - What data to use to answer certain questions. - Computers can be used to monitor supplies.	- Convert a series of multiple actions into a single loop. - Repeat actions initiated by the instructor. - Translate a picture program into a real-world dance. - Break down a long sequence of instructions into the smallest repeatable sequence possible. - Create a program for a given task which loops a sequence of commands. - Employ a combination of sequential and looped commands to reach the end of a maze. - Identify bugs in an existing program and assess how to correct them. - Identify the benefits of using a loop structure instead of manual repetition. - Break down a long sequence of instructions into the smallest repeatable sequence possible. - Identify bugs in an existing program and assess how to correct them. - Identify the benefits of using a loop structure instead of manual repetition. - Count the number of times an action should be repeated and represent it as a loop. - Create a program that draws complex shapes by repeating simple sequences. - Decompose a shape into its largest repeatable sequence. - Identify how to debug a program so that it will solve a puzzle.