

Year	Skills
Y1	<ul style="list-style-type: none"> Follow and give instructions using forward, backward and whole, half, quarter and three quarter turns. Control remote controlled and programmable toys using direction and turn. Predict the effect of a given instruction on a programmable toy. Plan and test a sequence of instructions Debug a sequence of instructions . Know that controlling a programmable toy is more precise than a remote controlled toy. Use an on screen resource to replicate movements of a programmable toy. Plan and test a sequence of instructions on screen. Use direction and turn cards to plan and record an algorithm to achieve a purpose using a remote controlled toy. Predict what a given algorithm will do related to a real life context Write their own algorithm relating to a real life context. Debug a given algorithm.
Y2	<ul style="list-style-type: none"> Plan and enter a sequence of instructions on a floor robot specifying distance and turn to achieve a given outcome. Debug a sequence of instructions. Understand the term sequence. Plan and test a sequence using distance and turn instructions to achieve a given algorithm. Find an alternative algorithm to one already given. Debug a program explaining why it needs to be changed. Edit a given algorithm to achieve a different outcome. Replicate an algorithm using programming software and debug Write an algorithm to produce a shape. Use repeat in a real life context. Predict what a given algorithm will do and test their predictions by creating a program using it.
Y3	<ul style="list-style-type: none"> Use logo type commands to control a floor robot. Understand how instructions given in a logo program relate to instructions given to a programmable robot / toy Solve problems with a floor robot and replicate their solutions on screen Use logo commands to write an algorithm and program e.g. to draw regular shapes. Explain what a given program does in a logo program and using a visual programming language. Debug a program written in logo commands and using a visual programming language. Use repeat in logo to write a program Test and debug given programs Write an algorithm using logo and using a visual programming language to achieve an outcome Explain how an algorithm solves a problem Write a program in which an object is used to trigger an action.
Y4	<ul style="list-style-type: none"> Test and improve given programs Improve efficiency in programs by comparing different solutions and by using repeat. Write and edit programs using logo commands. Write procedures using logo e.g. to draw polygons, letters and shapes. Use procedures as part of a program. Define variables e.g. to draw shapes on screen with logo and to create a score in a game. Plan and write a program using a flow chart structure. Use sensors to 'trigger' an action e.g. touching wall Write an algorithm and then create a program that will use a simple selection command for a game
Y5	<ul style="list-style-type: none"> Plan, debug and test algorithms and programs. Use looping and repeat until a condition is met in programs. Group commands to create procedures or sub-routines. Plan, write, debug and test programs using selection structures. Write programs in which an input controls an output and edit to give a different output. Create a program to simulate and control a real life system. Control on screen mimics and physical devices. Use 4 quadrants to identify position in a visual programming language. Use understanding of internal angles to program more complex shapes on screen. Write a program which uses more than one variable. Use a varying sensor as an input to trigger action in a program e.g. temperature or light.

Programming skills

Y6	<ul style="list-style-type: none">• Plan an algorithm using flow chart notation and then use it to write a program.• Write a program from a given algorithm to achieve a specified outcome. Use the program to test and improve the original algorithm.• Control on screen mimics and physical devices using more than one input and predict the outputs• Use selection structures in a program• Create variables in a program.• Use sensors to measure an input in order to trigger a sequence and procedure.• Edit programs using procedures / subroutines to improve efficiency.•
----	---