

## Progression in programming skill - alternative

Adapted from: Joint CAS / NAACE guidance - <http://www.computingatschool.org.uk/data/uploads/CASPrimaryComputing.pdf>

Year	Main activity	Support activity	Knowledge objectives
1	<p><b>Solving problems with Bee-Bots</b></p> <p>Enter sequence of instructions to navigate around a maze</p>	<p><b>Bee-Bots (app)</b></p> <p>Give a sequence of instructions to complete a simple task.</p>	<p>Understand what algorithms are</p> <p>Create simple programs</p>
2	<p><b>Move the turtle / Logo (with buttons)</b></p> <p>Generate a sequence of instructions including 'right angle' turns, followed by a range of shapes</p>	<p><b>Daisy Dinosaur</b></p> <p>Use the 'repeat' (loop) and 'when' commands, followed by planning instructions sequences for a sprite</p>	<p>Understand that algorithms are implemented as programs on digital devices</p> <p>Understand that programs execute by following precise and unambiguous instructions</p> <p>Debug simple programs</p> <p>Use logical reasoning to predict the behaviour of simple programs</p>
3	<p><b>Scratch – scripted animations</b></p> <p>Animate a number of sprites to interact with each other using a sequence of instructions</p>	<p><b>Lightbot</b></p> <p>Enter instructions to guide the robot around each maze, including the use of procedures</p>	<p>Write programs that accomplish specific goals</p> <p>Use sequence in programs</p> <p>Work with various forms of input</p> <p>Work with various forms of output</p> <p>Use repetition in programs</p>
4	<p><b>Scratch – Maths quiz</b></p> <p>User inputs the answer and produces an appropriate output using an 'if...then' statement. Additional questions are added, along with a score variable</p>	<p><b>Hopscotch (app)</b></p> <p>Use a range of inputs, loops and 'if... then' statements within a series of instructions</p>	<p>Design programs that accomplish specific goals</p> <p>Work with variables</p> <p>Design and create programs</p> <p>Debug programs that accomplish specific goals</p> <p>Use repetition in programs</p> <p>Use logical reasoning to detect and correct errors in programs</p>
5	<p><b>Scratch – Maze game</b></p> <p>Use the keyboard and infinite loops to control sprites. Use conditional statements to implement collision detection, lives and a score</p>	<p><b>FlowGo / Flowol / Lego WeDo</b></p> <p>Build and control a model using a sequence of instructions. Modify instructions based on sensor inputs.</p>	<p>Solve problems by decomposing them into smaller parts</p> <p>Use selection in programs</p> <p>Work with variables</p> <p>Use logical reasoning to explain how some simple algorithms work</p> <p>Use logical reasoning to detect and correct errors in algorithms</p> <p>Control or simulate physical systems</p>
6	<p><b>Kodu – Single player game</b></p> <p>Create a 3D digital world for a game with land, water and scenery. Program sprites to navigate their 3D world with an input, including the addition of dangerous items using conditional statements.</p>	<p><b>Cargo Bot (app)</b></p> <p>Use loops and conditions to refine algorithms</p> <p>Or: <b>Own game in Scratch / Alice</b></p>	<p><i>Reinforcing Y5 objectives with increased independence:</i></p> <p>Solve problems by decomposing them into smaller parts</p> <p>Use selection in programs</p> <p>Work with variables</p> <p>Use logical reasoning to explain how some simple algorithms work</p> <p>Use logical reasoning to detect and correct errors in algorithms</p>

