Progression in programming skill

Adapted from: primarycomputing.co.uk - http://primarycomputing.co.uk/a-computing-curriculum/

Year	Unit 1	Unit 2	Knowledge objectives
1	**Bee Bots** Give and follow instructions, which include straight and turning commands, one at a time. Explore outcomes when instructions are given in sequence. Give a simple sequence of instructions.	**Daisy Dinosaur / Bee Bots (app)** Discuss/explore what will happen when instructions are given in a sequence. Give a sequence of instructions to complete a simple task. Instructions use both movement commands and additional commands.	Algorithms are sets of instructions for achieving goals Algorithms can describe everyday activities and can be followed by humans and by computers. Computers need more precise instructions than humans do. Computers are controlled by a sequence of instructions. A computer program is like the narrative part of a story, and the computer's job is to do what the narrator says. Computers have no intelligence, and so follow the narrator's instructions blindly.
2	**Daisy Dinosaur** Use the 'repeat' (loop) and 'when' (conditional statement) command within a series of instructions. Plan a short 'story' for a sprite and write the commands for this. Edit/refine a sequence of commands.	**Move the turtle / Logo (with buttons)** Generate a sequence of instructions including 'right angle' turns. Create a sequence of instructions to generate simple geometric shapes (oblong /square). Discuss how to improve/change their sequence of commands.	Steps can be repeated within algorithms Algorithms can be represented in simple formats [storyboards and narrative text] Particular tasks can be accomplished by creating a program for a computer. Some computers allow their users to create their own programs.
3	**Hopscotch (app)** Use a variety of inputs Use the 'repeat' (loop) command within a series of instructions. Use the 'if… then' (conditional statement) command within a series of instructions	**Logo** Write a simple program in Logo to produce a line drawing. Use more advanced Logo programming, including pen up, pen down etc. Write a program to reproduce a defined problem, e.g. geometric shape/pattern.	Algorithms can be represented symbolically [flowcharts] or using instructions in a clearly defined language [turtle graphics]. Algorithms can include selection (if) and repetition (loops). Algorithms should be stated without ambiguity and care and precision are necessary to avoid errors. Algorithms are developed according to a plan and then tested. Algorithms are corrected if they fail these tests. A computer program is a sequence of instructions written to perform a specified task with a computer. Programs can be created using visual tools.
4	**Scratch *Racing car*** Navigate the Scratch programming environment. Create a background and sprite for a game. Add inputs to control their sprite. Use conditional statements (if… then) within their game.	**Kodu *Single player - free to navigate and avoid danger*** Create a 3D digital world for a game with land, water and scenery. Add a sprite to their world. Program their sprite to navigate their 3D world with an input. Use conditional statements ('if…then') to create dangerous items in their world.	Algorithms can include selection (if) and repetition (loops). Algorithms may be decomposed into component parts (procedures), each of which itself contains an algorithm. It can be easier to plan, test and correct parts of an algorithm separately. The idea of a program as a sequence of *statements* written in a programming language [Scratch] One or more mechanisms for *selecting* which statement sequence will be executed, based upon the value of some data item Programs can be created using visual tools. They can use a variety of control structures [selections and procedures].

5	**Scratch *The Ghostly woods*** Use external triggers and infinite loops to control sprites. Create and edit variables Use conditional statements **Cargo Bot (app)** Use loops and conditions to refine algorithms	**Scratch Robot Wars** Use variables to configure external outputs within Scratch Use external inputs to control external outputs Use conditional statements and infinite loops	Algorithms may be decomposed into component parts (procedures), each of which itself contains an algorithm. Algorithms can include selection (if) and repetition (loops). The behaviour of a program should be planned. One or more mechanisms for *selecting* which statement sequence will be executed, based upon the value of some data item
6	**Introduction to Python** Navigate Python programming environment Idle Declare variables Use a range of statements Use selection algorithms Use comparison and numerical operators	**Scratch *Temple Run*** Design their own game including sprites, backgrounds, scoring and/or timers. Their game uses conditional statements, loops, variables and broadcast messages. Their game finishes if the player wins or loses and the player knows if they have won or lost. Evaluate the effectiveness of their game and debug if required.	**Knowledge objectives:** Algorithms can be represented symbolically [flowcharts] or using instructions in a clearly defined language [turtle graphics] Algorithms are developed according to a plan and then tested. Algorithms are corrected if they fail these tests. Algorithms can include selection (if) and repetition (loops). A well-written program tells a reader the story of how it works, both in the code and in human-readable comments Computers can be programmed so they appear to respond ‘intelligently’ to certain inputs.