

Y6 COMPUTING – 4 CODING

QUESTION: HOW ARE FLOWCHARTS OR DESIGN DOCUMENTS USEFUL WHEN CODING?



Algorithms



Creating Media



Data & Information



Design & Development



Effective Use of Tools



Programming

WEEKS OF TEACHING	LINE OF ENQUIRY	THRESHOLD CONCEPT	SUBSTANTIVE KNOWLEDGE	DISCIPLINARY KNOWLEDGE	CURRICULUM ENDPOINT
1	Using Cloning	Code (motion, sound, sensing, variables and lists, operators)	- A clone in 2Code is a variable type that makes exact copies of an object. - An attribute is a detail that tells us more about something, e.g. we often have things like characters and objects, and each one has attributes that describe it. - A draggable attribute says whether the user of the program can drag an object when the program is running. - Repeat Over is a command in 2Code which repeats commands nested within it continuously as a loop. [Encourage children to add sound to their programs, including use of the piano tool in 'Choose a sound' and 'Record' to make their own short tunes or sound effects. Children can also use Audacity to create and upload their own sounds.]	- Set IF conditions for movements. Specify types of rotation giving the number of degrees. - Upload sounds from a file and edit them. Add effects such as fade in and out and control their implementation. - Use a range of sensing tools to control events or actions. - Use lists to create a set of variables. - Use the Reporter operators to perform calculations.	Design, write and debug programs that accomplish specific goals, including controlling or simulating physical systems; solve problems by decomposing them into smaller parts. Use sequence, selection, and repetition in programs; work with variables and various forms of input and output.
2	Using Hotspots	Code (motion, sensing)	A hotspot is a type of object that can be used to make a specific area of the screen react to actions such as objects entering or leaving the area.	- Set IF conditions for movements. Specify types of rotation giving the number of degrees. - Use a range of sensing tools to control events or actions.	Use logical reasoning to explain how some simple algorithms work and to detect and correct errors in

3	User Input	• Code (events, control, sensing, variables and lists, operators)	• Input is information going into the computer. This could be the user moving or clicking the mouse, or the user entering characters on the keyboard. On tablets, there are other forms such as finger swipes, touch gestures and tilting the device. • A variable is a named area in computer memory. A variable has a name and a value. The program can change this variable value. Variables are used in programming to keep track of things that can change while a program is running. In 2Code user-created variables can be text (strings), numbers or functions. • A datatype is the type of data, for example, a number or string, and what operations can be done with it. • Concatenation is the action of linking things together in a series.	• Set events to control other events by 'broadcasting' information as a trigger [JB note: breaking a program down into sub-routines/ functions and calling them]. • Use IF THEN ELSE conditions to control events or objects. • Use a range of sensing tools to control events or actions. • Use lists to create a set of variables. • Use Boolean operators [JB note: uses 1/O instead of true/false]. • Use the Reporter operators to perform calculations.	algorithms and programs. Select, use and combine a variety of software (including internet services) on a range of digital devices to design and create a range of programs, systems and content that accomplish given goals, including collecting, analysing, evaluating and presenting data and information.
4	Flowcharts and Control Simulations	• Code (motion, sound, events, control, sensing, variables and lists, operators)	• A flowchart is a diagram that uses specifically shaped, labelled boxes and arrows to represent an algorithm as a diagram. • A function is a named group of commands that a program can run when called by name, reducing the need to rewrite code repeatedly. • A simulation is a model that represents a real or imaginary situation. Simulations can be used to explore options and to test predictions. [Encourage children to add sound to their programs, including use of 'Record' in 'Choose a sound' and Audacity to create and upload their own sounds.]	• Set IF conditions for movements. Specify types of rotation giving the number of degrees. • Upload sounds from a file and edit them. Add effects such as fade in and out and control their implementation. • Set events to control other events by 'broadcasting' information as a trigger [JB note: breaking a program down into sub-routines/ functions and calling them]. • Use IF THEN ELSE conditions to control events or objects. • Use a range of sensing tools to control events or actions. • Use lists to create a set of variables. • Use Boolean operators [JB note: uses 1/O instead of true/false]. • Use the Reporter operators to perform calculations.	

<u>5</u>	Using Program Design Documents	• Code (motion, draw, events, control, sensing)	• The User Interface (UI) is the part of a computer program that you see and use on the screen. It is everything that helps the user tell the computer what to do and shows the user what the computer is doing. • Design in coding is a plan for the program showing the visual look of the user interface (the screen) with the objects. The algorithm can be represented as part of the design, showing actions and events. • Abstraction is a way of decluttering and removing unnecessary details to code the main task of the program. • Decomposition means breaking down a task into manageable components.	• Set IF conditions for movements. Specify types of rotation giving the number of degrees. • Combine the use of pens with movement to create interesting effects. • Set events to control other events by 'broadcasting' information as a trigger [JB note: breaking a program down into sub-routines/ functions and calling them]. • Use IF THEN ELSE conditions to control events or objects. • Use a range of sensing tools to control events or actions.	
<u>6</u>	Text-Based Adventures	• Code (events, control, sensing, variables and lists, operators)	• A text-based adventure is a computer game that uses text instead of graphics.	• Set events to control other events by 'broadcasting' information as a trigger [JB note: breaking a program down into sub-routines/ functions and calling them]. • Use IF THEN ELSE conditions to control events or objects. • Use a range of sensing tools (including proximity, user inputs, loudness and mouse position) to control events or actions. • Use lists to create a set of variables. • Use Boolean operators [JB note: uses 1/0 instead of true/false]. • Use the Reporter operators to perform calculations.	

ASSESSMENT OPPORTUNITY:

Can children create their own text adventure?

Children can confidently use 2Code to design and program interactive activities using variables, inputs, cloning, and hotspots. They can follow and create simple flowcharts and use them to debug and improve their programs. They understand how to build user interfaces and structure their code using functions and subroutines. They apply

decomposition and abstraction to break down and simplify tasks. They adapt code to personalise their work, including changing room content in a text-based adventure or modifying outputs based on user input. They show independence in debugging and generally create programs that meet the design specification