Unit 4.2 iProgram -iFunction

Year: Four (Level 2)
Term: Autumn

Background information for teachers

Advanced iFunction looks at creating programs to solve real world problems. Building upon their knowledge of 'Blockly' from Level 1 they will apply it to creating classic arcade games, control solutions and other problems mimicking real applications of programming. Pupils will also learn how to expand their code during the latter part of the course.

Learning Outcomes for the Unit

Throughout iProgram Level 2 we will be learning about 'Operators' and 'Syntax' through various coding tasks.

Key vocabulary
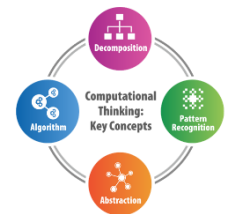
Computational Thinking	Computational thinking involves taking that complex problem and breaking it down into a series of small, more manageable problems.
Operators	In computer science, an operator is a character or characters that determine the action that is to be performed or considered.
Syntax	Syntax refers to the rules that define the structure of a language. Syntax in computer programming means the rules that control the structure of the symbols, punctuation, and words of a programming language.
Functions	Functions are "self contained" modules of code that accomplish a specific task.
Loops	In computer programming, a loop is a sequence of instruction s that is continually repeated until a certain condition is reached.
Variables	A named area in computer memory. A variable has a name and a value. The program can change this variable value.
Conditionals	Conditionals are expressions that evaluate to either true or false. They are mostly used to determine Program Flow through if statements and while loops.
If	A conditional command. This tests a statement. If the condition is true, then the commands inside the block will be run
Nesting	When you write a command inside something else e.g. a block of commands could be nested inside a timer.
If/Else	A conditional command. This tests a statement. If the condition is true, then the commands inside the 'if block' will be run. If the condition is not met, then the commands inside the 'else block' are run.
Debug/Debugging	Looking for any problems in the code, fixing and testing them.

Unit 4.2 iProgram -iFunction	
Learning Session 1	Advanced iFunction will start by recapping Level 1 going over the basics of algorithms, loops and computational thinking. Re-introduce Hopscotch and go over the basic functions. Set the class the challenge of creating their own versions of modern art using loops and randomness.
Learning Session 2	Using their knowledge of Hopscotch the class will create the first of two arcade games. This will be a Frogger game and will refresh their knowledge of conditionals and computational thinking.
Learning Session 3	Introduce written coding languages. This will be the students' first look at programming syntax, using Spirit box to introduce basic language as well as class and written tasks. We will view the benefits of written code over 'Blockly' style coding and apply written code to some simple problems.
Learning Session 4	Advanced variables. This lesson will look at how powerful a variable can be. We will discuss some problems and set small tasks for the class to complete to gain a greater understanding of variables and how they can be used to create more complex programs. The class will experiment with variables to create simple programs with separate characters that can interact with each other.
Learning Session 5	Application of coding skills. Giving the students a brief of a situation which they need to plan and build a program. Recap over computational thinking and looking at how they will approach the task, they will be challenged with building 'Space Invaders' (a classic arcade game). Pupils will have to break the task down into sections that will all have to interact with each other, using a combination of all of the skills they have learnt so far.
Learning Session 6	Present programs to the class to demonstrate what they have done to complete the brief. Use extra time to complete feedback and save screen shots of completed work
Learning Session 7	This session will run either as part of a longer half term or a contingency for pupils or classes who have exceeded the lesson plans for the half term. During this lesson pupils will learn to code variables into their games. Pupils may move on to adding randomisation and variables into different levels depending on their ability.