Bomere and the XI Towns Federation Knowledge Organiser - Computing

Topic: Programming B - Repetition in Games

Class/Year Groups: Stiperstones

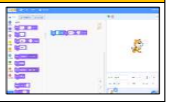
Term: Summer

What you already know?

This unit assumes that learners will have some prior experience of programming. The KS1 NCCE units cover floor robots and ScratchJr, and Scratch is introduced in the Year 3 programming units

The Basics of Scratch

-What is Scratch? Scratch is a website/ app that lets us code our own stories, games and animations.



-Scratch helps us to learn how to use programming language, whilst also being creative and using problem-solving skills.

There are three main areas in Scratch:

-The Blocks Palette (on the left) contain all of the different blocks: puzzle piece commands which control the animation.

-Code Area (in the middle) is where the blocks are placed to create a program.

-Stage with Sprite (right) is where the output of the program is presented. The sprite is the character.

Attributes: There are three attributes of the sprite which we can change to make our animation: Code, Costumes, Sounds.

-Event Blocks: Event blocks are coloured yellow and are used to sense different events that happen e.g., the green flag being clicked.

-Action Blocks: Action blocks include 'Motion' blocks, 'Sound' blocks and 'Looks' blocks. They make the sprite move, make sounds and change appearance.

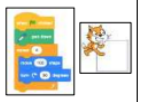
What you will learn:

Loops and Repetition

-Pen Drawing in Scratch: Select the 'add extension' icon in the bottom left corner. Then select 'pen.' This allows you to draw with your sprites.

-The Repeat Block: Select 'code' and then the 'control' blocks (orange). Here you will find the repeat block. It should be placed around the command blocks that you want to repeat. The number of times something is repeated can be typed into the white area.

-Creating Shapes: Selecting 'pen down' (in the 'operators' blocks) can be followed by use of the motion blocks to determine the line that will be drawn (e.g. 'move 10 steps'). Turning a number of degrees changes the direction of the pen. Placing the repeat block around this motion code can allow more complex shapes to be drawn.



-Count-Controlled/Infinite Loops: We can control the number of 'loops' of a command with the number typed into the 'repeat' block. The 'forever' block makes a command continue infinitely (forever).

Event Managing and Efficiency

-We should ensure that programs are coded and labelled in easy-to-understand, user-friendly ways.

-Using the 'events' blocks logically can help to make your programming easy to use. E.g. when 's' key pressed a square is drawn, when 'h' key is pressed a hexagon is drawn.

-Efficiency is about getting the right result in the easiest way possible, wasting little time or effort. Our use of the repeat and loop tools should help to create efficient programs.

Algorithms, Trialling, Debugging

-Designing an **algorithm** (set of instructions for performing a task) will help you to program the sequence that you require.

-Programmers do not put their computer programs straight to work. They **trial** them first to find any errors:

-Sequence errors: An instruction in the sequence is wrong or in the wrong place.

-Keying errors: Typing in the wrong code.

-Logical errors: Mistakes in plan/thinking.

-If your algorithm does not work correctly the first time, remember to **debug** it.

Vocabulary:

programming	when we make a set of instructions for computers to follow.
Scratch	a program that we can use in order to code our own stories, animations and games.
algorithms	a set of instructions to perform a task
attributes	there are three attributes of the sprite which we can change to make our animation: Code, Costumes, Sounds.
event blocks	are coloured yellow and are used to sense different events that happen e.g., the green flag being clicked.
action blocks	include 'Motion' blocks, 'Sound' blocks and 'Looks' blocks. They make the sprite move, make sounds and change appearance

National Curriculum Objectives:

- 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
- Use logical reasoning to explain how some simple algorithms work, and to detect and correct errors in 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