Enriching lives every day; enabling our school community to learn, achieve and flourish through living 'life in all its fullness'

Innovation

Subject: Computing

Year group: 5

Term: Spring Term

Unit name: Selection in Physical Computing

Vocabulary	
Refine procedures	Iteration (repetition)
Variable	And, if, then , else
Problem solving	Two-way selection
True/false	Program
Change inputs	Algorithm
Different outputs	Count-controlled loop
Open ended problem	Condition
Abstraction	Command block

Big idea: To make use of prior knowledge of repetition and conditions to write algorithms which use the concept of selection (through the if, then structure)

This unit allows children to use physical computing to explore the concept of selection by programming a controller known as a Crumble controller. They connect and program components (output devices- LEDS and motors) . Children use skills in repetition and conditions and are introduced to selection to produce a program.

Progression of skills :

Understands the difference between, and appropriately uses if and if, then and else statements.

Understands how the If... Then... Else structure can be used to select different outcomes depending on whether a condition is true or false.

Writes programs that ask questions and use selection to control the outcomes based on the answer .

Designs, writes and debugs modular programs using procedures.

Prior learning:

Year 3 Sequence in Music. Children used motion, sound and event blocks to create a representation of a piano in the Scratch environment .
Year 4 Repetition in shapes Children have used repetition and loops with Purple Mash 2Logo.

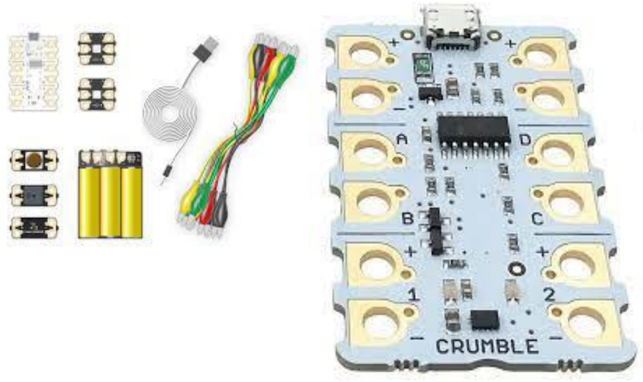
Future learning:

Year 6 Variables in Games This unit uses Scratch to explore variables to simulate a scoreboard.
Year 6 Sensing This unit teaches children how to use block programming using a BBC Microbit. This brings together the repetition, selection and variables.

National Curriculum links :

Computing

- 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



Implementation:

Children become familiar with the Crumble Controller. They control a simple circuit connected to a Computer using a continuous loop.

Use additional devices (such as LED) connected to the Crumble controller and write a program using repetition and count controlled loops.

They learn that a loop can stop when a condition is met. They use a switch as an input and write a program that uses a do until loop to control a repeating light pattern.

Conclude that a loop can be used to repeatedly check whether a condition has been met . Children are introduced to selection using the If…Then…structure. They use it to direct the flow of a program .

They design a physical project which includes selection.

They create the algorithm needed to create a controllable system. Test, evaluate and debug algorithm.

Subject knowledge

The NCCE unit can be completed in full this term. There are a number of videos provided that demonstrate concepts and are provided on the slides. It is recommended that teachers watch these and model to the children with a live demonstration during lessons.

Crumble programming software uses the same drag and drop style as Scratch. You will need to write programs that turn on and off LEDS (sparkles), change LED colours, spin motors, use push switches as inputs and combine a number of these peripherals. Additionally, you will be connecting the Crumble controller with battery packs, sparkles, motors and push switches. For further support on using Crumbles try the "Crumble Getting Started Guide" at https://redfernelectronics.co.uk/Crumble-getting-started/.

Repetition

You will need to know that repetition is used in programming to give the same instruction or set of instructions several times. Repetition uses loops as the means to give these instructions. This unit makes use of two types of loops: infinite and count-controlled. These have been defined below.

Infinite loop

An infinite loop is a loop that commands the instruction/set of instructions to repeat forever. When an infinite loop is used in a program there is no way of ending the program as the command(s) within the loop will be repeated endlessly. For this reason, infinite loops should only be used when writing a program that is intended to run forever. The exception to this is when using selection in physical computing as you will see throughout this unit.

Count-controlled loop

A count-controlled loop is a form of repetition in which a set of commands are carried out a specific number of times. Count-controlled loops should only be used when it is known how many times a set of commands need to be repeated.

Condition-controlled loop

A condition-controlled loop is a form of repetition in which a set of commands stop being carried out when a condition is met. The condition could be anything from when 'score' in a game reaches a certain value or when a key on a keyboard has been pressed.

Conditions

Conditions are statements that need to be met for a set of actions to be carried out. They can be used in algorithms and programs to control the flow of actions. When a condition is met it is referred to as true and when it is not met it is referred to as false. You need to be able to identify and use conditions in algorithms in the form of statements to both start and stop sets of action. Additionally, you need to understand that conditions can be used in loops and when they are, that the set of actions in the loop will be carried out repeatedly until the condition is true. For example, 'until button A is pressed'.

Selection

When designing programs, there are often points where a decision must be made. BBC Bitesize defines selection as:

> Selection - "A decision within a computer program when the program decides to move on based on the results of an event." KS3 Bitesize

These decisions are known as selection, and are implemented in programming using IF statements. Selection is used to control the flow of actions in algorithms and programs by checking if a condition (see above) has been met. If it has been met, the identified actions will be carried out. When selection is used in programs, loops (see above) have to be used to instruct the device to check the condition repeatedly. Without using loops the condition would only be checked once. In the Crumble programming software, selection is implemented through the If__ Then__ command block.