

Teacher glossary

# Introduction

**Use this glossary to build your confidence and fluency in the key language needed to teach computing.**

Find our downloadable student glossary for use in the classroom on [Classroom resources](https://microbit.org/teach/classroom-resources/)



# Abstraction

The process of focussing on the key information and ignoring the information that is not important for a specific context.

When pupils use the LED outputs to create animations they will use their abstraction skills to consider how they can best represent their item. They will need to identify the characteristics that are unique to that item and consider how these can be represented on a 5 x 5 LED set.

# **Algorithms**

A set of step by step instructions that can be followed by humans to achieve a specific goal. Usually written to be implemented as a program on a computer. When using the inputs and outputs of the micro:bit, pupils can be presented with a purposeful reason for writing algorithms and see the links between these and the commands they will use when implementing this as a program.

# **Collaboration**

The process of working with a partner to create, test and debug an algorithm or computer program.With multiple input options, the micro:bit offers the opportunity for two pupils to each have their own idea turned into a script and executed within the same program.

# **Communication**

When computers and hardware are connected together to form a network, they are able to communicate with each other. Data stored on one computer can be displayed on another.

Using the radio functions, data (in the form of words or numbers) can be sent from one micro:bit and displayed using the LEDs on another micro:bit or can be used to control the output of another micro:bit.



# **Control**

Generating an output that is carried out in the real world when a change in a condition is sensed. Through the use of micro:bit’s inputs and outputs, real-life situations can be replicated to give pupils purposeful programming scenarios. Furthermore, the use of micro:bit as a prototype can open up a wider world of possibilities.

# **Debugging**

The process of finding and fixing errors in a computer program. By combining everyday inputs, easily identifiable outputs and an accessible block-based programming environment, pupils are supported in identifying which aspect of their program did not execute correctly and can then locate the area of the program where the bug(s) may exist.

# **Decomposition**

Breaking a problem or an overall action into smaller parts to make the overall process more manageable. Pupils can use decomposition to identify what has to be constructed for the input and output part of the program when controlling real-world devices or creating prototypes.

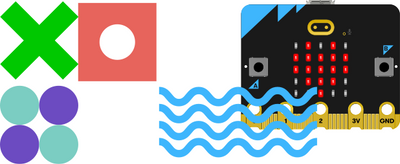
# **Encryption**

Encryption allows for data sent between devices to be scrambled into incomprehensible text. To decipher the message both the sender and the receiver need to have shared a cryptographic key usually in the form of numbers.

Encryption can be explored by sending messages between two (or more) micro:bits accompanied by a numerical value. The messages will only be displayed if the number sent matches the agreed value of the cryptographic key.

# **Evaluation**

Making judgements on the success of an algorithm or a program in achieving the goals set. The use of everyday inputs and readily identifiable outputs allow pupils to judge if they have constructed a program that meets the requirements of the task.



# **Input**

The way by which data is sent to a computer. With its buttons, sensors and accelerometer, the micro:bit can be used to construct programs that respond to inputs without the requirement of attaching additional hardware.

# **Iterative design**

The process of developing a product through a cycle of analysing, making, testing, evaluating and improving. The micro:bit can support pupils’ use of iterative design as it can be used as a prototype for many of their solutions to real-world problems. This will give pupils a greater appreciation of the roles of testing and evaluation as well as the importance of refining and persevering with their ideas.

Computer systems are the hardware and software that the computers need to run and the peripherals that support this. There is a great deal of variation in what computer systems look like. Regardless of their appearance, all computers work by following the input, process, output model.

The micro:bit can develop pupils’ understanding of what a computer is and allow them to appreciate the breadth to which computers are used in their daily life.

# **Logical Reasoning**

Using existing knowledge and understanding to explain why something will happen. These use of everyday inputs such as movement, temperature, light and noise and the outputs such as the LEDs and sound support children in making predictions on what will happen when a program is executed.

# **Networks**

Networks are formed when computers are connected together so that data can be shared between these computers. The micro:bit can be connected to another micro:bit (or several others) allowing data to be shared through the radio functions.

# **Output**

The way that data generated by a computer is presented to the user. micro:bit has a range of outputs such as the 5x5 LED grid and the speaker (micro:bit v2). These provide excellent engagement for pupils when constructing programs.



# **Patterns**

Identifying commonalities between algorithms and programs, and using previous experience to inform the program. As the micro:bit can do a lot from a few commands, pupils will have plenty of experience in writing and refining programs. They can use these experiences to apply their existing knowledge across a range of contexts in turn deepening their understanding.

# **Process**

What the computer does with the data that has been inputted. When constructing programs, pupils will explore a range of ways that inputted data can be processed such as creating LED patterns or playing music.

# **Program**

A set of instructions written for a computer to follow to achieve a specific goal.When writing programs for the micro:bit, pupils have the opportunity to apply their knowledge of the concept they are studying in an engaging and real-life manner.

# **Repetition**

A key computer concept: an instruction or set of instructions that need to be carried out on more than one occasion can be repeated rather than written out again. These instructions can be repeated several times or endlessly.

To create ongoing animations, pupils will need to consider how they use repetition when deciding how many times their animation will be repeated. Infinite repetition in the form of forever loops is essential when using selection to ensure that the micro:bit is constantly checking if a specified condition has been met.



# **Selection**

A key computing concept: an instruction or set of instructions is/are only carried out if a certain condition is met.With its range of input sensors, the micro:bit offers plenty of opportunities for pupils to explore selection commands using a range of conditions based on the different inputs. Pupils can construct programs that result in multiple outputs.

# **Sensors**

A device that detects or records changes in the environment. The micro:bit’s thermometer, light meter and microphone mean pupils can use sensors to monitor environmental conditions and construct outputs that occur when a change is sensed. This allows pupils to see the links between the programs they are writing and those used to respond to events in the real world.

# **Sequence**

A key computing concept: instructions need to be given in the order that we want them to be carried out. When creating animations, pupils need to understand that images need to be programmed in the order in which they are to be displayed. They also gain an appreciation for the use of delays when using ‘pause’ to allow sufficient time for an image to be displayed.

# **Variables**

Variables are pieces of data stored when a program is running. This data can be recalled, changed and used. The micro:bit provides excellent opportunities to use variables in a range of contexts particularly when using inputs to set, change and display data. Furthermore, understanding can be deepened through the use of the data generated by the sensors.