

Barefoot - litter hunt

**Lessons:** 2

**Programming languages:** MakeCode

**Target age:** 7-11 yrs

**Subjects & topics:**

* Computer systems: Input/output
* Programming: Variables

# Unit of work summary

Lessons aimed at students aged 9-11. Students use buttons and sensors on the micro:bit to make different images display, are introduced to the concept of variables, and create a device that records details of litter that can be recycled in the local area. You will need to use physical micro:bits and battery packs for these lessons.

## Overall key learning

* I can explain what the buttons and sensors on the micro:bit do
* I can program sensors and buttons using the MakeCode editor
* I can store data with variables using the MakeCode editor
* I can design a counting device using the micro:bit
* I can program a counting device using the MakeCode editor
* I can use a counting device to record data

## Additional skills

Collaboration, evaluation, presenting, problem-solving

## Lesson 1: Barefoot - litter hunt, lesson 1

In this lesson, pupils develop their understanding of the micro:bit by using the buttons and sensors on the device to make different images display. Pupils are introduced to the concept of a variable to store data, then create a program to alter the value of a variable using the buttons and sensors. Pupils go on to create a ‘push the button’ game, where they race against another player to see who can press the button the fastest.

**Key learning:**

* I can explain what the buttons and sensors on the micro:bit do
* I can program sensors and buttons using the MakeCode editor
* I can store data with variables using the MakeCode editor

## Lesson 2: Barefoot - litter hunt, lesson 2

In this lesson, pupils remind themselves of how a variable functions within a program. They are then introduced to the problem from an environmental organisation, which requires them to record details of litter in the local area, including identifying litter that can be recycled. Pupils design and develop their program, then take the micro:bit out of the classroom to record details of the litter and undertake a litter pick.

**Key learning:**

* I can design a counting device using the micro:bit
* I can program a counting device using the MakeCode editor
* I can use a counting device to record data

# Curriculum links

These lessons are mapped to the following learning objectives and standards:

## England National Curriculum

#### KS2 computing curriculum

Curriculum aims:

* can analyse problems in computational terms, and have repeated practical experience of writing computer programs in order to solve such problems

Students should be taught to:

* 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
* 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

[Read the full KS2 computing curriculum.](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/239033/PRIMARY_national_curriculum_-_Computing.pdf)

#### Year 4 science curriculum

Students should be taught to:

* Living things and their habitats - recognise that environments can change and that this can sometimes pose dangers to living things

[Read the full KS2 science curriculum](https://assets.publishing.service.gov.uk/government/uploads/system/uploads/attachment_data/file/425618/PRIMARY_national_curriculum_-_Science.pdf)

## Scotland Curriculum for Excellence

#### Technologies

* I can extend and enhance my knowledge of digital technologies to collect, analyse ideas, relevant information and organise these in an appropriate way. (TCH 2-01a)
* I can explain core programming language concepts in appropriate technical language. (TCH 2-14a)
* I understand how information is stored and how key components of computing technology connect and interact through networks. (TCH 2-14b)
* I can create, develop and evaluate computing solutions in response to a design challenge (TCH 2-15a)

[Read the full Curriculum for Excellence: technologies](https://education.gov.scot/Documents/Technologies-es-os.pdf).

## Northern Ireland Curriculum - Primary

#### Using ICT - Computational Thinking and Coding – Level 4

Pupils should:

* create a more sophisticated coding project using a broad range of commands; and/or
* solve a given problem using commands in a programming environment

#### Programmable devices (micro:bit)

* look at and talk about examples of coding projects, including the use of motion, looks, lights or sounds, sensors, control and events such as ‘if...then’ and ‘loop until’ (or equivalent) that make the code more efficient **(also includes variables at Level 5)**;
* recognise that these projects are composed of different components and break the task into smaller manageable tasks (decomposition);
* in small groups, plan and storyboard their own coding project, working out what different parts of the program must do, using logical reasoning to discuss and compare the commands that are required for their algorithm and predict the outcome;
* use a range of commands to create a project including triggering commands such as ‘if...then’ and ‘loop until’ to facilitate a more efficient method of interaction **(also includes variables at Level 5)**;
* test and debug at regular intervals and collaborate with others to solve problems as they arise.

Pupils should:

* share their work (possibly using digital tools), respond to feedback and comment on others’ work; and
* organise files and export work in an appropriate format so that others may view it.

[Read the CCEA's Using ICT Desirable Features](https://ccea.org.uk/learning-resources/using-ict-desirable-features)

#### The World Around Us - Change over Time - KS2

Pupils should be enabled to explore:

* The effects of positive and negative changes globally and how we contribute to some of these changes (Science and Technology - Recycling)

[Read the CCEA's World Around Us KS2 curriculum](https://ccea.org.uk/downloads/docs/ccea-asset/Curriculum/Key%20Stage%202%20Statutory%20Requirements%20for%20The%20World%20Around.pdf)

## Curriculum for Wales

#### Science and technology

Progression step 2 - computation is the foundation for our digital world:

* I can create simple algorithms and am beginning to explain errors.
* I can follow algorithms to determine their purpose and predict outcomes.

Progression step 3 - computation is the foundation for our digital world:

* I can use conditional statements to add control and decision-making to algorithms.
* I can identify repeating patterns and use loops to make my algorithms more concise.
* I can explain and debug algorithms.

Progression step 2 - design thinking and engineering offer technical and creative ways to meet society's needs and wants:

* I can produce designs to communicate my ideas in response to particular contexts
* I can explore how different component parts work together.

Progression step 3 - design thinking and engineering offer technical and creative ways to meet society's needs and wants:

* I can draw inspiration to design from historical, cultural and other sources.
* I can use design thinking to test and refine my design decisions without fear of failure.

[Read the full science and technology curriculum](https://hwb.gov.wales/curriculum-for-wales/science-and-technology/descriptions-of-learning/)

#### Digital competency framework

Progression step 2 - data and computational thinking - problem-solving and modelling:

* I can break down a problem to predict its outcome.
* I can detect and correct mistakes which cause instructions (a solution) to fail (debug)
* I can change instructions to achieve a different outcome.
* I can identify repetitions or loops in a sequence.

Progression step 3 - data and computational thinking - problem-solving and modelling:

* I can create and refine algorithms and flowcharts to solve problems, making use of features such as loops, Boolean values and formulae.
* I can understand the importance of the order of statements within algorithms.

Progression step 2 - producing - evaluating and improving digital content:

* I can give an opinion about my own work and suggest improvements based on the success criteria

Progression step 3 - producing - evaluating and improving digital content

* I can explain reasons for the layout and content of my own work and the work of others
* I can ensure my output is appropriate for specific purposes

[Read the digital competence framework](https://hwb.gov.wales/curriculum-for-wales/cross-curricular-skills-frameworks/digital-competence-framework)

## USA CSTA Standards

#### Grades 3-5

* 1B-CS-01 - Describe how internal and external parts of computing devices function to form a system.
* 1B-CS-02 - Model how computer hardware and software work together as a system to accomplish tasks
* 1B-CS-03 - Determine potential solutions to solve simple hardware and software problems using common troubleshooting strategies.
* 1B-DA-06 - Organize and present collected data visually to highlight relationships and support a claim.
* 1B-DA-07 - Use data to highlight or propose cause-and-effect relationships, predict outcomes, or communicate an idea.
* 1B-AP-16 - Take on varying roles, with teacher guidance, when collaborating with peers during the design, implementation, and review stages of program development.

[Read the CSTA Standards in full.](https://csteachers.org/k12standards/ )

This content is published under an [Open Government](https://www.nationalarchives.gov.uk/doc/open-government-licence/version/3/) licence.