**Title**: Bee-Bot ruler

***SUB HEADING***: Program a robotic device to follow a path

**Summary Text:** Explore the concept of sequencing steps, using Bee-Bots to measure length.

**Year Level**: F–2



<Credit line> Ars Electronica Center/flickr

<Alt text> Photograph of a Bee-Bot, from the front, showing red and green arrow keys on its back.

# Suggested steps

1. Support students to create their own Bee-Bot ruler (15 cm) and demonstrate how far the Bee-Bot moves when programmed to move forwards or backwards one step.
2. Provide a set of vertical and horizontal lines of different lengths. (Draw a set of vertical and horizontal lines of different lengths on large sheets of chart paper and place these on the floor. Alternatively, use the sides of objects or classroom furniture as items to be measured, eg bookshelf, desks/tables, doorway.)

**Visual Impairment:**

Students with limited or no vision can participate by having one hand on the Bee-bot and one on the table/ground to feel and hear how far the Bee-bot moves in one step. They can use their outstretched thumb and index finger, or similar, to predict the length of various physical objects. After students estimate length of an object using their fingers/hands, they use the Bee-bot to see if their estimate was correct.

1. Invite students to estimate the length of the various lines or pieces of furniture. The Bee-Bot rulers may be used to help students estimate the lengths.
2. Record the estimates and program the Bee-Bot to travel the estimated length. Students may need support calculating increments of 15 cm.



**<Alt text>** Image of a named 'Bee-Bot Ruler' with the line 'A Bee-Bot moves 15 cm with each step running underneath the name, and between an arrow pointing left and an arrow pointing right.

1. Working in pairs, students can continue to estimate and measure other lengths around the classroom or school.

# Discussion

* Compare estimates with actual measurements.
* Ask 'What functions does the Bee-Bot have that makes it a useful tool for measuring length?'
* Consider ways to program the Bee-Bot to follow the path of a two-dimensional shape. What extra instructions would the Bee-Bot need to undertake this task?
* Through discussion and examples, support students in understanding why is it important to give the Bee-Bot a precise set of instructions.

# Why is this relevant?

This activity is an authentic way to introduce students to simple programming while consolidating measurement concepts. It focuses on developing foundational skills in computational thinking, and on developing an awareness of digital systems through personal experience of them.

F–2 students should be provided with opportunities to explore new concepts such as algorithms through guided play, including hands-on, kinaesthetic and interactive learning experiences. Students begin to develop their design skills by conceptualising algorithms as a sequence of steps/procedures for carrying out instructions to solve simple problems or achieve certain things, such as identifying steps in a process or controlling a Bee-Bot.

At the F–2 level, where learning at the pre-programming stage is the expectation, there is no requirement to learn a particular programming language. However, in years F–2 students do learn some basic programming skills, such as working out steps and decisions required to solve simple problems. For example, they program a robotic toy or sprite to move in a certain direction. The focus at this level is on designing a sequence of steps.

# Assessment:

## Student self-assessment task

Consider the commands used to move the Bee-Bot. Students complete a simple cloze activity to evaluate understanding:

Bee-Bots travel \_\_\_\_\_\_cm each time they move one step. The arrow keys at the front and back are used to program the Bee-Bot to move \_\_\_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_\_\_\_\_. The arrow keys on each side are used to program the Bee-Bot to move \_\_\_\_\_\_\_\_\_\_ or \_\_\_\_\_\_\_\_\_\_\_\_. I can start a new program by pressing the \_\_\_\_\_\_\_\_\_\_\_ button.

# Australian Curriculum Alignment

## Mathematics

### Measurement and Geometry

**Using units of measurement**

Use direct and indirect comparisons to decide which is longer, heavier or holds more, and explain reasoning in everyday language [(ACMMG006)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACMMG006)

Measure and compare the lengths and capacities of pairs of objects using uniform informal units [(ACMMG019)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACMMG019)

Compare and order several shapes and objects based on length, area, volume and capacity using appropriate uniform informal units [(ACMMG037)](http://www.australiancurriculum.edu.au/mathematics/curriculum/f-10?layout=1#cdcode=ACMMG037&level=2)

**Location and transformation**

Describe position and movement [(ACMMG010](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACMMG010)

Give and follow directions to familiar locations [(ACMMG023)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACMMG023)

## Technologies – Digital Technologies

Follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems [(ACTDIP004)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACTDIP004)

Recognise and explore digital systems (hardware and software components) for a purpose [(ACTDIK001)](http://www.australiancurriculum.edu.au/curriculum/contentdescription/ACTDIK001)

## ICT Capability

Typically, by the end of Year 2, students:

**Select and use hardware and software**

Identify and safely operate a selected range of appropriate devices, software, functions and commands when operating an ICT system and attempt to solve a problem before seeking help

es. Students begin to develop their design skills by conceptualising algorithms as a sequence of steps/procedures for carrying out instructions to solve a problem or achieve something, such as identifying steps in a process or controlling the Bee-Bot.

At the F–2 level, where learning at the pre-programming stage is the expectation, there is no requirement to learn a particular programming language. However, students do learn some basic programming skills such as working out steps and decisions required to solve simple problems. For example, they program a robotic toy or sprite to move in a certain direction. The focus at this level is on designing a sequence of steps.