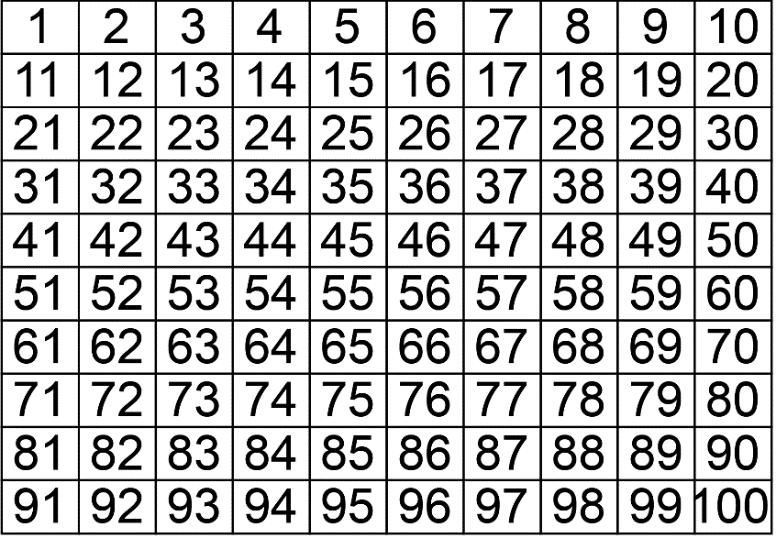
# Skip counting with Bee-Bots

*Please refer to the online lesson plan on the DT Hub to access all website links and additional resources.*

Program a robotic device to follow a path

**Year Level**: F–2 :Play a skip counting game where students program the Bee-Bot to stop at multiples of a set number, eg 2, 4, 5, 10 on a number grid.



<Alt text> A number grid arranged in rows 10 x 10

# Suggested steps

1. Use a number board to practise counting number sequences to and from 100. Skip count by twos, fives and tens, both forwards and backwards. Students use the number sequences to program the Bee-Bot to start on zero and stop on each multiple, using a correct sequence, eg multiples of 5 (5,10,15,20,25,30).
2. Differentiate to suit learners’ ability by increasing the complexity of the task, eg start from any starting point; use other number sequences.

# Discussion

* Consider the factors that led to success or failure. Emphasise the importance of clear and precise instructions.
* Ask why it is important to test and retest programs.

# Why is this relevant?

This activity is an authentic way to introduce students to simple programming while consolidating number 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 the 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 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.

# Assessment

## Teacher assesment

**Use a simple checklist to evaluate achievement.**

Can the student:

* describe the functions of the Bee-Bot?
* design a correct sequence of instructions?
* select the appropriate buttons to program a Bee-Bot to follow a simple path?
* describe the purpose of programming a Bee-Bot?
* use directional language correctly?

# Australian Curriculum Alignment

## Technologies – Digital Technologies

Processes and Production Skills:

* Identify and explore digital systems and their components for a purpose (AC9TDI2K01)
* Follow and describe algorithms involving a sequence of steps, branching (decisions) and iteration (repetition) (AC9TDI2P02)

## Mathematics

Algebra:

* Recognise, continue and create pattern sequences, with numbers, symbols, shapes and objects, formed by skip counting, initially by twos, fives and tens (AC9M1A01 )

Number:

* Quantify sets of objects, to at least 120, by partitioning collections into equal groups using number knowledge and skip counting (AC9M1N03 )

Location and transformation:

## Describe the position and location of themselves and objects in relation to other people and objects within a familiar space (AC9MFSP02 )

## Give and follow directions to move people and objects to different locations within a space (AC9M1SP02 )

## 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