Buzzing with Bee-Bots

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

Learning hook

Sequence game – whole class activity

1. Give students a floor mat or large surface divided as a grid. The grid can be of any dimension, for example 3 x 3; 4 x 4; 5 x 5, 10 x 10. (A mat with a grid marked out, or some masking tape on the floor to show the grid. Make sure it is highly visible and that the squares are large enough for students to stand or sit in.)

****To further create engagement, you could print out checker/chess boards, have students choose a token (either individually or in pairs), and follow a set of directions.

For those with limited, low, or no vision, ice-cube trays or similar with tactile grid lines could be used instead of a checker/chess boards. Complexity can be increased by joining two ice-cube trays to make a larger grid.

1. Invite a student to volunteer to act in role as a robot. The robot’s task is to locate and retrieve an object placed somewhere on the grid. The robot must start from the bottom left hand corner.
2. Explain that the robot needs to be given a set of instructions in order to retrieve the object. The robot understands words like Forward or Reverse and symbols like arrows and F for Forward, R for Reverse, etc.
3. Invite the students to write a set of instructions for the robot to follow. For example:

FFRR or ↑↑→→
(Forward 2 squares then move right 2 squares)

**** You may also wish to model an example set of instructions on the whiteboard to make sure students understand what is expected of them.

1. Share student responses and direct the robot to move according to the sequence. Student acting in role as the robot must follow the directions given.
2. Repeat the activity once or twice, using different volunteer robots and placing the object to be retrieved in a different location each time.
You many choose to increase or decrease the size of the grid.

For background notes about coding, programming and computational thinking download this PDF

Learning map and outcomes

Discuss the importance of instructions. Explain that digital technology systems are not magic: they follow instructions.

* Share the learning intentions for the lesson.

For example: Today we will be:

* learning to write a set of directions
* directing a robot (Bee-Bot) to move in different directions
* learning to use directional words (Forward, Reverse, Left, Right)

Learners have a clear understanding of what they are learning and can see how it fits into bigger context.

WILT chart displayed on the wall (What I am Learning Today).

Learning input

Model the control buttons on a Bee-Bot.

Encourage students to spend some time playing with the Bee-Bots to work out a sequence of steps needed to:

* move the robot from one location to the other
* move to create the perimeter of a square.

Learning construction

Provide students with challenges. For example:

* Code the Bee-Bot to move forward for a specific distance in cm.
* Code the robot to move across a number chart and skip count.
* Code the Bee-Bot to trace differently sized squares.
* Code the robot to move across a world map from one country to another.
* Invent a Bee-Bot challenge. Learners record their learning, for example, by:

**Extension Task:** making a video recording with commentary about the algorithm they have used (for example using the Explain Everything app)

* recording a simple written algorithm they have used (which could be done collaboratively in something like a Google Doc).

Short video explanation of Explain Everything

Learning demo

Sharing solutions – whole class activity

1. Encourage learners to share their solutions to the coding challenges. Ask learners to describe the series of steps they used to program their Bee-Bot.
2. Record the solutions.
3. Ask students to suggest directional words and/or symbols they used as they participated in the challenges. List these words.

Curriculum Links

**Links with Digital Technologies Curriculum Area**

Knowledge and Understanding

* Recognise and explore digital systems (hardware and software) for a purpose (AC9TDIFK01) .
* Identify and explore digital systems and their components for a purpose (AC9TDI2K01).

Processes and Production Skills

* Follow, describe and represent a sequence of steps and decisions (algorithms) needed to solve simple problems (ACTDIP004)

**Links with other Learning Areas**

English

Language (Text structure and organisation)

* Understand that different types of texts have identifiable text structures and language features that help the text serve its purpose (ACELA1463 ).

Mathematics

Measurement and Geometry (Location and transformation)

* Describe position and movement (ACMMG010 ).

**Assessment rubric**

The rubric below is based on the SOLO Taxonomy which is a great framework to measure the depth of learner understanding against a set of learning outcomes. There are many different rubric types you can choose, you will find different examples on this site Whichever rubric scoring system you decide to use, ensure it focuses on learner progress and provides an opportunity for feedback to learners which goes beyond a simple numeric score. There is much research on effective feedback on a summative task and Dylan William suggests that comment only feedback is the most effective, as learners otherwise only focus on a numeric score and do not heed the advice in the feedback.

|  | **Quantity of knowledge** | **Quality of understanding** |
| --- | --- | --- |
| **Criteria** | **Pre-structural** | **Uni-structural** | **Multi-structural** | **Relational** | **Extended abstract** |
| **Algorithmic thinking** | Learner shows no evidence of understanding an algorithm. | Learner is able to say what an algorithm is. | Learner is able to follow and execute an algorithm. | Learner is able to explain the causes of each element of their algorithm. | Learners are able to generalise their understanding of an algorithm to create their own. |
| **Use of Bee-Bots** | Learner is unable to make the Bee-Bot work. | Learner is able to make the Bee-Bot complete one command. | Learner is able to make the Bee-Bot complete multiple commands. | Learner is able to string together multiple commands to complete a task/algorithm. | Learners are able to design more complex tasks for the Bee-Bots to complete beyond the scope of the tasks given (innovate).  |
| **Optional score** | 0 | 1 | 2 | 3 | 4 |