

Subject: Computing	Year: 2 – Spring 1 – Programming A – Robot Algorithms
<p>National Curriculum objectives</p> <ul style="list-style-type: none"> • Understand what algorithms are, how they are implemented as programs on digital devices, and that programs execute by following precise and unambiguous instructions; • Create and debug simple programs; • Use logical reasoning to predict the behaviour of simple programs. 	
<p>To begin this unit, the children should have already learnt:</p> <p><u>Year 1</u> Programming is when we make a set of instructions for computers to follow. Robots, such as floor robots like Bee-bots, are one type of machine that can follow programs. Floor robots have buttons which help us to direct them. We can use algorithms (a set of guidelines to perform a task) to program floor robots along routes.</p>	<p>The learning in this unit will prepare the children to learn these things in the future:</p> <p><u>Year 3</u> <i>ScratchJr</i> is a programming environment with three main areas: the Blocks Palette; Code Area; and Stage with Sprite.</p> <p><u>Year 4</u> Logo is a text-based programming language, where we can type commands which are then drawn on the screen. Instead of typing in the code to create each individual shape, we can save time by repeating a sequence of instructions. We use the ‘repeat’ function and create ‘infinite’ or ‘count-controlled’ loops.</p> <p><u>Year 5</u> Microcontrollers control real-life objects (like LEDs and motors) through the construction of programs. Conditions are a means of controlling the flow of actions in a program.</p> <p><u>Year 6</u> A variable is something that is changeable. A variable can be set and changed throughout the running of a program. Programmers will apply the Use-Modify-Create model: learners will experiment with variables in an existing project, then modify them, before they create their own project.</p>
<p><u>Key Enquiry Question</u></p> <p>Why does the language of instructions need to be clear and precise? What will happen if these (instructions) are given in the wrong order? What algorithm have you made? What do you predict to be the outcome of this (algorithm)? What will your robot be able to do in your program? Can you identify any mistakes in the example algorithm?</p>	<p><u>The Big Idea:</u></p> <p>Programming is when we make a set of instructions for computers to follow and we use algorithms (a set of instructions to perform a task) to help robots to do things that we want them to. Debugging can help to correct algorithms and programs.</p>

To achieve ARE, pupils will need to be secure in the following knowledge:

By the end of this unit, children will know:

- A series of instructions is a sequence;
- What happens when we change the order of instructions;
- A series of instructions can be issued before they are enacted;
- You can predict the outcome of a program.

Vocabulary:

Programmed; algorithm; button; direction; forward; backward; robot; left; right; route (introduced in Y1).

Obstacle; design; chunking; error; debugging.

By the end of this unit, children will be able to do:

- Choose a series of words that can be enacted as a sequence;
- Choose a series of instructions that can be run as a program;
- Create a program;
- Trace a sequence to make a prediction;
- Run a program on a device;
- Debug a program they have written.

Useful Resources:

[Teaching Programming to 5- to 11-year-olds](#)


Bee-bot floor robots

COMPUTING: PROGRAMMING KNOWLEDGE ORGANISER

Overview



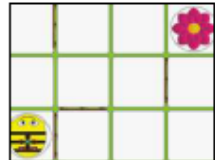
Robot Algorithms

- Programming is when we make a set of instructions for computers to follow.
- Robots are one type of machine that can follow programs - they follow what we instruct them to do.
- We use algorithms (a set of instructions to perform a task) to help robots to do things that we want them to.
- Debugging can help to correct algorithms and programs.



Algorithms and Instructions


- **Algorithms:** Algorithms are precise set of instructions, that a computer can turn into a code. A floor robot has a computer inside of it.
- **Programs:** When we press the buttons of our floor robot, we are creating a program for it to follow. The program is how the algorithm is run as code on the robot.
- **Instructions:** It is important that our instructions to the floor robot are clear. If our sequence of instructions is in the wrong order, has anything missing, or has anything additional, the floor robot will end up in a different place! Plan the route to avoid obstacles and get to the right place.

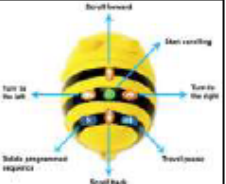
Using a Floor Robot



- **Robots:** Robots are machines that we can program to do human jobs.
- Robots help us to do things, for example to help us clean, mow and learn!
- Robots in factories make things, and in hospitals they help make us better.



-Turning on a Bee-bot:
Before we use a Bee-bot, we need to make sure it is charged. To turn it on, using the switch underneath. You can tell that the Bee-bot is on because its eyes light up. Remember to switch it back off again after you have finished using it.



-Buttons: Bee-bots have buttons on the top. They each make the Beebot do something different (see picture).
-The arrows move the Bee-bot in different directions. The GO button makes the Bee-bot start its program. The X button makes the Bee-bot forget the last set of instructions.



Designing Algorithms	Chunking and Debugging
<p>-We can buy or create mats for floor robots. We then need to design our algorithms so that the robot follows the given route.</p> <p>-We should carefully consider the start point & end point that we want the robot to reach.</p> <p>-Use symbols (e.g. arrows, crosses) to indicate the commands that will be inputted as a program.</p>  	<p>-Chunking: With larger programs, we can break the task into chunks and create algorithms for each chunk.</p> <p>-Debugging: Debugging is finding and fixing errors in our algorithms and programs. These errors can include:</p> <ul style="list-style-type: none"> -<u>Sequence errors:</u> An instruction in the sequence is wrong or in the wrong place. -<u>Keying errors:</u> Typing in the wrong code. -<u>Logical errors:</u> Mistakes in plan/thinking.

Important Vocabulary

Program	Robot	Algorithm	Direction	Route	Obstacle	Design	Error	Chunking	Debugging
---------	-------	-----------	-----------	-------	----------	--------	-------	----------	-----------