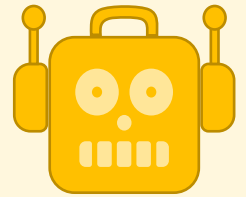


Spirit of Innovation

STEAM Resources



Technology

Year One

Algorithms

All systems checked and ready to go

Links

- 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

Resources

- Bee-Bots
- Blank Bee-Bot mat
- Or large piece of paper
- Bee-Bot picture cards ([Download](#))
- Bee-Bot jacket ([Download](#))
- Whiteboards
- Whiteboard pens
- Coding Sheet ([Download](#))
- Debugging Sheet ([Download](#))

Skills

- Working in a team
- Working as a pair
- Discussing ideas
- Writing own simple code
- Follow/write instructions
- Program a Bee-Bot
- Record an algorithm

Questions

- What do the buttons on the Bee-Bot do?
- How could you write go forwards 3, turn left?
- Where will the Bee-Bot start?
- What order will you be visiting each picture? Why?
- What do we call computer instructions?
- What does an algorithm designer do?
- What does a coder do?
- What is debugging? Why is it important?

Activity

Activity One

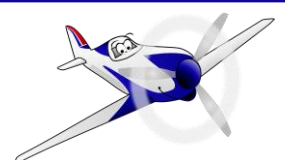
Whole Class/pairs

(20 - 30 mins)

In the classroom/Outside area/Hall



PIONEERS OF POWER



Bring the class to carpet. Start by being 'bossy,' instruct certain pupils to do a task e.g. stand up, get a white board, sit down, walk to the door, get a pencil, come back to your carpet place, sit down. Explain to make things happen you require a sequence of instructions. Explain that computers have to follow instructions too but these instructions are called algorithms.

Tell the class that they are going to be learning about programming and algorithms. Describe how algorithms are step-by-step instructions to make something happen.

Divide the class into pairs; explain that one will be called an algorithm designer, the one that thinks of the instructions; whilst the other one will be a robot.

The algorithm designer will provide the robot with 3 basic verbal instructions (forwards/backwards/turn left/turn right/stop); e.g. move forwards 3, turn left, move, backwards 5. Swap roles.

This is a fun game and can be repeated during playtime or as a fun 5-minute filler.

Explain that robots/computers and other similar technology will only do what the algorithm need it to do. Therefore, the instructions (algorithms) need to be clear, simple and correct.

Activity Two

Whole Class/pairs
(20 -30 mins)

Sit the class in a circle: -

Introduce the Bee-Bot using a real Bee-Bot and perhaps even a Bee-Bot image on the IWB. Look at the various buttons.

What do you think each button will do?

Upon receiving answers, the teacher/pupil can press the button. Then placing the Bee-Bot down the class will see if the answer is correct.

Once the pupils are familiar with each button a challenge can begin; "how we could get the Bee-Bot to follow a set route?"

In partners, discuss how to get the plane (Bee-Bot) to the take-off area and fly between two post?

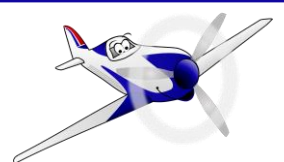
Using whiteboards record simple instructions; using their own notations. Ask various pairs to explain their notations, plus use their instructions to program the Bee-Bot. Teacher to correct any mis-conceptions and praise appropriate notations at this point.

Ask the class if anyone can remember what we call the person who writes instructions for a computer? (algorithm designer)

Explain that the person who programs the algorithm, pushes the buttons on the Bee-Bot, is called a *coder*.



PIONEERS OF POWER



Activity Three

In small groups mixed ability pairs
(20 – 30 mins)

It is expected that pupils have already had experience of Bee-Bots and have some ideas about how they work. If not allow pupils to play with Bee-Bots first before starting this activity or start with Activity Two.

This activity involves planning algorithms and using them to program a Bee-Bot around a themed Bee-Bot mat.

In pairs, plan a route for the Bee-Bot to navigate. Create a sequence of instructions (an algorithm) to guide the Bee-Bot to specific points on the Bee-Bot mat. Record this route, on a whiteboard.

Using the instructions on the whiteboard, program the Bee-Bot.

Watch the Bee-Bot as it travels, does it take the route as planned? If not, what needs to be altered? Any corrections are called *Debugging*. Debugging is the process of detecting and correcting the errors in a program.

Create another route that starts and ends at the same place, but is different than the first one? However, this time one plans and writes the route on the whiteboard; they will be known as the algorithm designer. Then their partner, the coder, programs the Bee-Bot to move along the route. Both can look for *Bugs* in the program, correct these bugs before moving on.

Swap roles, so each pupil can be the algorithm designer and the coder.

Activity Four

Pairs
(10 – 20 mins)

Coding and Debugging [Download both sheets](#)

In pairs look at the Bee-Bot map; ask the pupils to see if they can get the Bee-Bot to the charging point, but need to avoid all obstacles on the way. Discuss the buttons that they would need press on the Bee-Bot, in order for it to follow their chosen route. Pupils to write this algorithm onto the grid below the map. Once recorded use the algorithm to program the Bee-Bot again; check it does take the route that is written. If not 'debug' the code and correct the algorithm.

In pairs look at the algorithm that is written below the Bee-Bot map. Pupils use this algorithm to program the Bee-Bot. Once programmed they will need to watch the Bee-Bot at the same time as following the algorithm; looking out for the BUG! Once debugging the program pupils will need to correct the 'bug' and then check they are correct by reprogramming the Bee-Bot with the new algorithm.



PIONEERS OF POWER

