

Procedures

Ages: 11-14

Length: 1 hour

Equipment: PDF print out

Introduction

This lesson introduces the relevant words; **procedure**, **functions** and **comments** through activities that link to student's lives to help them relate to the concepts introduced. It teaches students to identify procedures and demonstrate their use in programming. This lesson will build upon prior learning relating to algorithms.

Curriculum Alignment	✔ UK National Curriculum Computing Key Stage 3
Learning Objective	✔ To identify a procedure ✔ To demonstrate the use of procedure
Keywords	✔ Procedure ✔ Functions ✔ Comments
Resources	✔ Procedure and Algorithm steps list ✔ Completed Procedure and Algorithm steps list
Lesson Sections	✔ Setting the Scene ✔ Activity 1 – Identifying procedures ✔ Activity 2 – MiRo simulator ✔ Summary

Setting the Scene

Do you need step by step instructions to complete a task all the time?

You complete tasks daily and don't need to know how to do it every time.

Concept	How it is used
<p>A <u>procedure</u> is a series of steps that are stored to be called upon within the main code.</p> <p>A <u>procedure</u> performs a task where as a function returns data.</p> <p><i>Can you think of a time where you have learnt what to do and when you went to do the same task again, you did not need instructions to complete it?</i></p>	<p>When you write code, it can save time if you have a set of instructions that can be used whenever it is called upon.</p> <p>A <u>procedure</u> is a set of instructions that can be actioned within the main code by the name given to it.</p>

You can create many procedures alongside the main code with a name given to it to call upon in the main code. Within Blockly, both are created used the block 'function'.

First let us look at an example of what a **procedure** is.

Activity 1

Activity 1 is focused on getting students to think about what a **procedure** is and relate this to their everyday tasks.

Scenario 1	Scenario 2
<p>You are making a cake and the instructions ask you to add flour and mix with margarine and egg. There are separate procedures for:</p> <ul style="list-style-type: none">☹ How to add enough flour needed?☹ How to add margarine needed?☹ How to crack an egg and add it to the mix	<p>You want to play on your Xbox, how do you know what to do when you log on?</p> <ul style="list-style-type: none">☹ What is the log on process?☹ How do you access a game?☹ How do you use the controller within the game?

Can you think of 2 other procedures you might follow in a normal day?

Small Group Activity

Split the students into small groups of 3 or partners, and give them a list of tasks (either the simple tasks below or of your choice).

Open the door

Brush your teeth

Go and have a shower

Make breakfast

Do your homework

Press the power button

Pick up your bag

Create a poster

In your group, I want you to look at the list of tasks in front of you.

You need to split them into two groups:

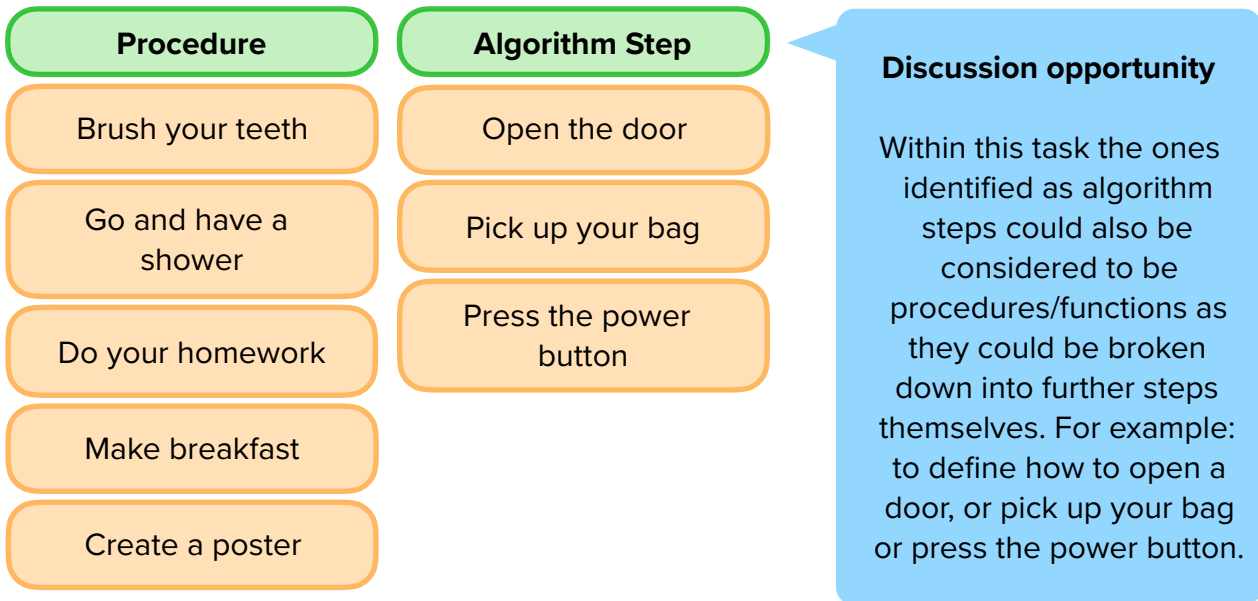
1. Procedures
2. Algorithm Steps

Encourage discussion within the group, ask them to think about the tasks they have done today and where they have had to follow instructions and where they knew what to do without guidance.

As humans you learn from the tasks you complete, when a task is learnt, like how to read and write, the knowledge is stored to be called upon when you next need it to read a book or write a letter.

Completed Small Group Activity

Here is the sorted arrangement for the provided blocks.



Differentiation

- 🦋 If pupils need extra support give them with one procedure 'brush your teeth' and ask them how do they know how to brush their teeth? They are taught and then remember this in order to complete the task when told to brush their teeth.
- 🦋 For higher ability pupils ask them to add three new procedures and three new algorithm steps and compare with other groups.

Additional Small Activity

This will guide students to create a function in Python. At the start of the code, you need to use 'def' followed by the name given to the function. All the code to be placed within the function is indented within it.

```
def aFunction():  
    #code goes here
```

The function is then called upon within the main code whenever the following code is added:

```
aFunction()
```

Below is the code for creating a function to output the words 'Welcome MiRo':

```
def aFunction():  
    print("Welcome MiRo")
```

What would the output be in the following code?

<u>Code</u>	<u>Output</u>
<pre>def aFunction(): print("Hello MiRo") aFunction() aFunction()</pre>	<p>Hello MiRo</p> <p>Hello MiRo</p>

Activity 2

What blocks are available for *procedures*? They are located within the **'functions'** tab.

We name the *procedure* in the section with the default text **'do something'**.

It is good practice when writing any code, to add **comments** where possible to remind yourself what the code is to do or to allow other people to collaborate on a coding project. They can then see the description or notes on the parts of code completed.

Click the blue question mark to access the comments area to describe the function you are creating. Within python programming a comment is written with a '#' before it to define it as a comment and not part of the program.

How can we get MiRo to use a *procedure* to facilitate facial movements? We are going to **create a program to define the procedures to blink**. Think about the steps you would need to consider to create a MiRo blink. How might this be written out...

Why would commenting within code help a project?

It helps other people such as project partners or teachers understand your code.

How can we get MiRo to use a *procedure* to facilitate facial movements? We are going to **create a program to define the procedures to blink**. Think about the steps you would need to consider to create a MiRo blink. How might this be written out...

Can you write some instructions out for the algorithm to blink below?

1. Loop instructions 2-5
2. Close eyelids
3. Wait 1 second
4. Open eyelids
5. Wait 1 second

What would the name of your procedure be?

Blink

Why is the name of the procedure just as important as the content?

Because it will help remind you of the purpose of the procedure as you get further down the code program.

This activity is based on MiRoCODE. Guide students to their computers and to start a new workspace either individually or in pairs. It instructs students in how to create and run programs using procedures. Students are now guided step by step on how to create the algorithm above, this may change due to what they have written.

Students are guided to create + run an algorithm to get MiRo to blink and twitch ears.

For advanced students, the **Step-by-Step** guide can be left off and they can work their way through the steps on their own.

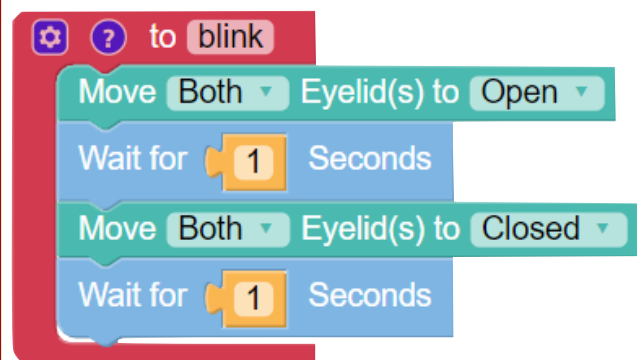
Activity 2 - Program solution - Part 1 Two Functions for blinking and ear twitching

NB. There can be more than one solution in programming, encourage experimentation and use solution below as a guide

Step 1-6

- 🦋 Add 'to do something' from 'Functions'
- 🦋 Set the function name to 'blink'
 - Add a comment to state it is a function to make MiRo blink.
- 🦋 Add 'move left eyelid(s) to open' block from 'Simple Motion'.
- 🦋 Connect into 'blink' function.
 - Set to both eyes.
- 🦋 Add 'wait for 1 seconds' block from 'Control'.
- 🦋 Connect to 'move both Eyelid(s) to Open' block.
- 🦋 Add 'move left eyelid(s) to open' block from 'Simple Motion'.
- 🦋 Connect to 'wait for 1 seconds' block
 - Set to both eyes and closed.
- 🦋 Add 'wait for 1 seconds' block from 'Control'.
- 🦋 Connect to 'move both Eyelid(s) to Open' block.

** Encourage students to use right click and duplicate instead of finding the blocks again.

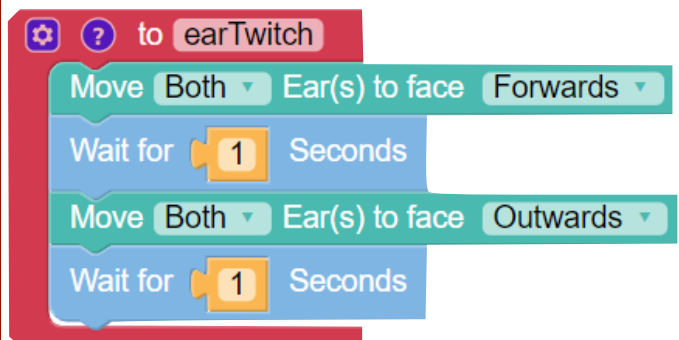
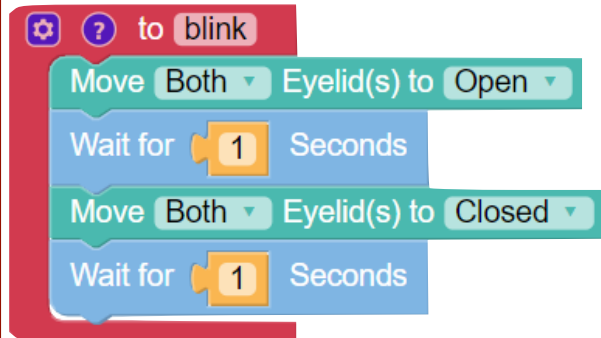


** Opportunity to discuss what the function is doing. At the moment nothing would happen as the function is created but not called upon within the main program.

Step 7-10

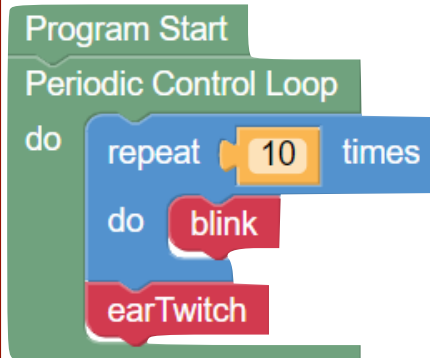
- 🦋 Add 'to do something' from 'Functions'
 - Rename the function to 'earTwitch'
- 🦋 Add 'move left Ear(s) to face forwards' block from 'Simple Motion'.
- 🦋 Connect into 'earTwitch' function
 - Set to both ears.
- 🦋 Add 'wait for 1 seconds' block from 'Control'.
- 🦋 Connect to 'move both Ear(s) to face forwards' block.
- 🦋 Duplicate both blocks within the function 'earTwitch'
- 🦋 Connect in same order to 'wait for 1 seconds' block.
 - Set to 'move both Ear(s) to face outwards'.

** Duplication can come from any area of the workspace.



Step 18

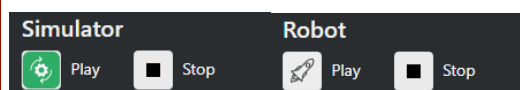
- 🦋 Add 'Program Start' from 'Control'
- 🦋 Add 'Periodic Control Loop do' block from 'Control'
- 🦋 Connect together.
- 🦋 Add 'repeat 10 times' from 'Loops'
- 🦋 Connect into the 'Periodic Control Loop' block.
- 🦋 Add 'blink' from 'Functions'
- 🦋 Connect into the 'repeat 10 times' block.
- 🦋 Add 'earTwitch' from 'Functions'
- 🦋 Connect in after the 'repeat 10 times' block.



Step 18

- 🦋 Click Simulator Play OR
- 🦋 Click Robot Play

** If you are using the physical MiRO ensure the IP address is correctly added to the onscreen code to allow the code to be communicated to MiRO to action.



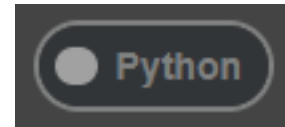
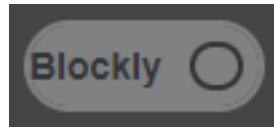
** If you are using the simulator remember to zoom in to see MiRo's face.

Activity 2 - Program solution - Part 2 Moving into Python

Step 1

- 👉 To flip to look at the python code, click the 'blockly' button at the top of the white canvas.
- 👉 When the canvas is switched the button will have the word 'python' on it and is pressed again to return to block code.

** Opportunity to highlight to students that **functions are always defined at the top of a program.**



Can you find the two functions?

Located at the top of the code.

What code defines a function?

Always defined with 'def' at the start followed by the functions name.

Step 2-4

- 👉 Locate the lines in each function controlling the time delay between actions
- 👉 Edit the time set within the code from '1' to '0.5'
- 👉 There are 4 x locations to edit.

robot.sleep(1) to robot.sleep(0.5)

```
5 # definitions
6
7 # Describe this function...
8 def blink():
9     robot.set_joint(miro.constants.JOINT_EYE_L, 0.0)
10    robot.set_joint(miro.constants.JOINT_EYE_R, 0.0)
11    robot.sleep(0.5)
12    robot.set_joint(miro.constants.JOINT_EYE_L, 1.0)
13    robot.set_joint(miro.constants.JOINT_EYE_R, 1.0)
14    robot.sleep(0.5)
15
16 # Describe this function...
17 def earTwitch():
18    robot.set_joint(miro.constants.JOINT_EAR_L, 0.0)
19    robot.set_joint(miro.constants.JOINT_EAR_R, 0.0)
20    robot.sleep(0.5)
21    robot.set_joint(miro.constants.JOINT_EAR_L, 1.0)
22    robot.set_joint(miro.constants.JOINT_EAR_R, 1.0)
23    robot.sleep(0.5)
```

Step 10-14

- 👉 Run your program.
- 👉 Experiment with the settings and investigate.
- 👉 Change the time further up and down to observe the change in output.

Summary

Have a discussion with the class about what they have learnt in the lesson. Discuss the new words learnt: **procedures**, **functions** and **comments**, and talk through any difficulties they had.

Ask students to complete the self-assessment and can be done by thumbs up, down and centre or using the images; on the following 3 questions.

Questions

Can you identify a function in block based code and python?

Can you describe what a procedure is?

Can you create a use a function to control MiRo's movement?