# TuringLab Farmbot Mapping (Secondary)

## Overview & Mapping – Programme of Study (Computing)

| Lesson | Lesson Objectives | Programme of Study for Computing statements |
|---|---|---|
| 1 | Understand<br>• what is meant by a command<br>• what is meant by a computer program<br>• what is meant by a text editor<br>Use<br>• the `move_forward` function to move the farmbot forward | 3.1, 3.3, 3.6, 3.7 |
| 2 | Understand<br>• what is meant by calling a function<br>• what arguments are and how to pass them to functions<br>• what is meant by an algorithm<br>• what is meant by a string<br><br>Use<br>• the `prepare_soil` command to get the farmbot to prepare a square of soil<br>• the `return_home` command to return the farmbot to it's charging station<br>• the `plant` command to control the farmbot to plant a crop | 3.1, 3.2, 3.3, 3.4, 3.6, 3.7 |
| 3 | Understand<br>• what is meant by a string<br>• what is meant by sequence<br>• the importance of syntax: quotes, brackets<br>Identify<br>• a function call within a computer program<br>Use<br>• a string as an argument in a function<br>• the `turn` command to rotate the farmbot | 3.1, 3.3, 3.6, |

| | | |
|---|---|---|
| 4 | Understand<br>• what is a function call<br>• how a sequential algorithm can modify the virtual farmbot world<br>Identify<br>• an argument within a computer program<br>• a string within a program<br>• an algorithm that performs a simple task 'to prepare soil and then plant a crop'<br>Use<br>• syntax to complete a python program<br>• an algorithm based upon a simple task<br>• an algorithm that requires sequential understanding<br>• an algorithm that requires use of python syntax and farmbot commands | 3.1, 3.2, 3.3, 3.4, 3.6 |
| 5 | Understand<br>• the importance of naming conventions: camel and snake case<br>• the importance of using the exact spelling for function names: capitalisation, dashes<br>• what is meant by a bug in coding<br>• what is meant by debugging in coding<br>• what is meant by a console and why it is useful<br>• the difference between a syntax, logic, name and indentation errors<br>Identify<br>• snake-case function names<br>Modify<br>• code to remove any bugs<br>• code to remove several bugs from a computer program<br>Create<br>• a computer program without causing bugs | 3.1, 3.2, 3.3, 3.4, 3.6 |
| 6 | Create an algorithm without mistakes that will plant three rows of crops | 3.1, 3.2, 3.3, 3.6 |
| 7 | Understand<br>• the difference between poorly and properly commented code<br>• what is meant by a comment and how to use them to increase code maintainability<br>Identify<br>• the correct code that harvests a row of crops<br>• a line of code which will not run due to a comment | 3.3, 3.6 |

| | | |
|---|---|---|
| | Modify<br>• the flow of control in a buggy computer program to fix the code<br>Use<br>• comments to add structure to a computer program<br>• the `harvest_crop` command to control the farmbot to harvest crops<br>• the `ship_crops` command to ship crops collected by the farmbot<br>Create<br>• algorithms to complex sequential tasks | |
| 8 | Understand<br>• common error messages<br>• what is meant by debugging<br>Identify<br>• the use of the snake-case naming convention<br>• the correct bug provided a computer program<br>• the correct algorithm for a given solution<br>Modify<br>• a computer program to remove a bug<br>Use<br>• comments to correctly comment a computer program<br>Create<br>• an algorithm using sequential commands | 3.1, 3.2, 3.3, 3.6 |
| 9 | Create an algorithm to plant a checkerboard pattern of crops. | 3.1, 3.2, 3.3, 3.6 |
| 10 | Create an algorithm to harvest alternating rows of tomatoes and aubergines. | 3.1, 3.2, 3.3, 3.6 |
| 11 | Understand<br>• what is meant by a for loop<br>• what is meant by a Python Keyword: for<br>• what indentation is and when to use white space<br>• what is meant by iteration<br>• when to use a for loop to create better code<br>Use | 3.1, 3.2, 3.3, 3.4, 3.6 |

| | | |
|---|---|---|
| | • a for loop to write more succinct code<br>• White space to correctly write a for loop<br>Create<br>• an algorithm using a for loop | |
| 12 | Create an algorithm that uses multiple for loops to control the Farmbot | 3.1, 3.2, 3.3, 3.4, 3.6 |

Overview & Mapping – Teach Computing Curriculum Year 7 Programming Unit

| Label | Teach Computing Curriculum Statement | Covered in Turing Lab – Farm Bot | Farm Bot |
|---|---|---|---|
| PS | Use an IDE to write and execute a Python program. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| PS | Locate and correct common syntax errors. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| CS | Call functions and use the results they return in expressions. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| PS | Use variables to keep track of information. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| PS | Trace through branches and loops and sketch state. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| CS | Use selection (if) to control the flow of program execution. | N/A | |
| DTAS | Create lists and access individual elements | N/A | |

| | | | |
|---|---|---|---|
| PS | Trace through programs that manipulate lists. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| DTAS | Perform common operations on lists. | N/A | |
| DTAS | Access individual string elements (characters). | L2, L3, L4 | |
| CS | Use iteration (while) to control the flow of program execution. | L11, L12 | |
| DTAS | Perform common operations on strings. | N/A | |
| PS | Use variables to keep track of counts. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| PS | Trace through programs that iterate over sequences using for. | L11, L12 | |
| CS | Use iteration (for) to iterate over lists. | N/A | |
| PS | Use variables to keep track of sums. | N/A | |
| PS | Combine features to develop solutions to meaningful problems. | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | |
| CS | Use iteration (for) to iterate over strings. | N/A | |
| | N/A | L2, L4, L6, L7, L8, L9, L10, L11, L12 | Create Functional Algorithms |
| | N/A | L1, L2 | Understand what a program is. |
| | N/A | L1, L2 | Understand what a command is. |
| | N/A | L1, L2, L3, L4, L5, L5, L6, L7, L8, L9, L10, L11, L12 | Use python code to manipulate objects in an IDE. |
| | N/A | L2, L4, L6, L7, L8, L9, L10, L11, L12 | Use Computational Thinking to decompose problems. |

PS = Programming Skills

CS = Control Structures

DTAS = Data Types and Structures