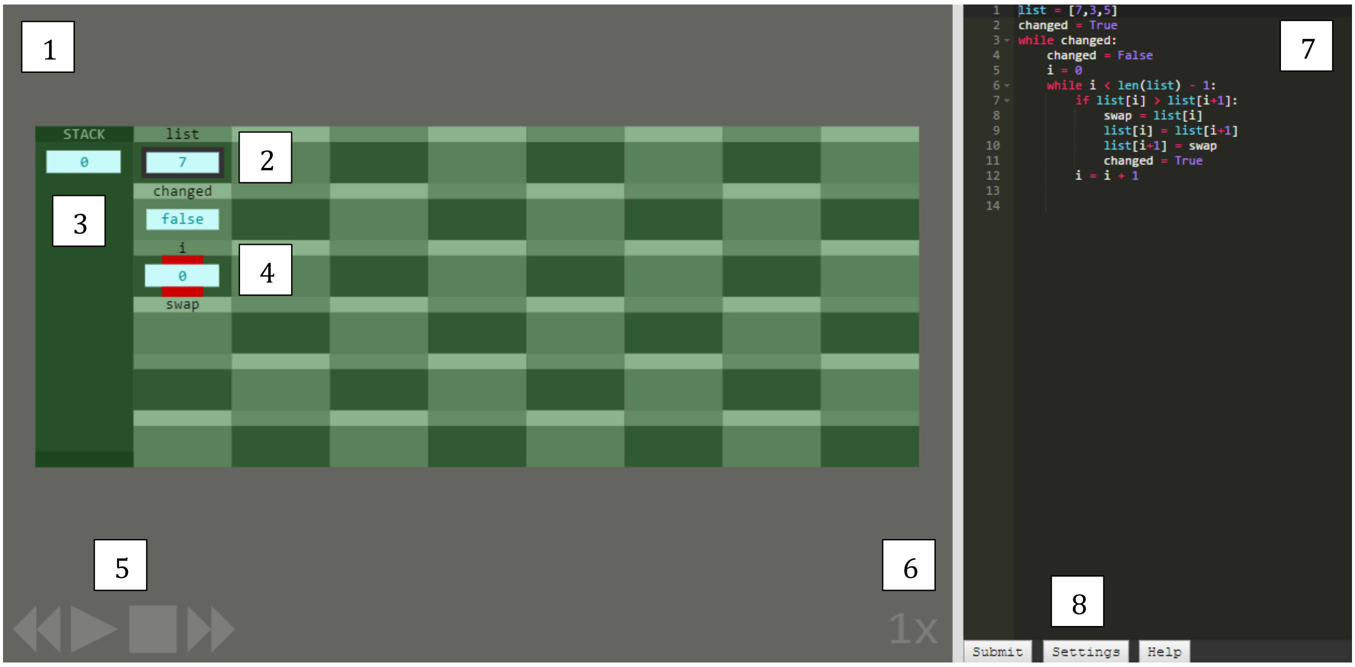
USER MANUAL

INTRODUCTION TO INTERFACE



```
list = [7,3,5]
changed = True
while changed:
    changed = False
    i = 0
    while i < len(list) - 1:
        if list[i] > list[i+1]:
            swap = list[i]
            list[i] = list[i+1]
            list[i+1] = swap
            changed = True
        i = i + 1
```

ID	Function	ID	Function
1	Playback area	**5**	Playback controls
2	List value	**6**	Playback speed
3	Stack	**7**	Code entry area
4	Robot	**8**	Submission buttons and settings

1 - PLAYBACK AREA

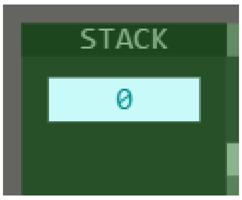
This is the area within which the playback of the visualisation of your program is displayed. This playback is made up of the following components…

2 – LIST VALUE

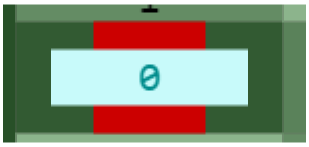
This is a value that appears on the grid. It is stored in the variable named “list” and its first most value is a 7. This variable can have different values assigned to it and as a list is made up of multiple sub-values it can also have different values assigned to it including other lists.

3 – STACK

This is the area within which the current operands are stored. For example if you request an addition of two variables whose values are 3 and 5 and assign it to a third variable, the robot will go to the first variable and add 3 to the stack and then to the second variable to add 5 to the stack. It will then operate on the two values placing an 8 on the stack and then go and place it form the stack onto the third variable.



4 – ROBOT



This is the robot that the stack uses to execute its instructions such as collecting values form variables, performing operations and assigning values to variables. The last value to be put on the stack will be displayed as though it is held by the robot

5 – PLAYBACK CONTROLS

These are the controls that control playback speed, they are the same as typical playback controls consisting of fast forward, rewind, play and stop.

6 – PLAYBACK SPEED

This displays the current playback speed.

7 – CODING AREA

This is the code entry area; it is simply an embedded version of Ace Editor.

8 – SUBMISSION BUTTONS

These are the buttons used to submit your code as well as bring up the settings panel and the help panel.

CODING FEATURES

ASSIGNMENT

Variables can be assigned in all the ways that python would typically allow:

```
a = [0,1,2]                         b = “hello”
a[0] = 1                            c = b[0]
```

A value can be assigned to any variable. You may individually assign indexes of lists but not strings, however, you can get a character in a string by index.

OPERATIONS

The operations that are implemented in this solution are as follows:

```
a = b + c          a == b          a = (b + c)/(d + e)
a = b – c          a != b          not a
a = b * c          a < b           a and b
a = b / c          a > b           a or b
```

STATEMENTS

The statements that are implemented in this solution are as follows:

```
if a == b:                         for a in [0,1,2]:
    [statements]                         [statements]

while a == b:
    [statements]
```

BUILT-IN FUNCTIONS

The built-in functions that are implemented in this solution are as follows:

```
len(iterable)   -> length of the iterable
str(value)      -> string version of the value
int(value)      -> integer version of the value
float(value)    -> float version of the value
bool(value)     -> boolean version of the value
```

SETTINGS

The settings provided include colour and the width and height of the grid on the screen. By increasing width and height you can gain more spaces to place variables and colour is purely aesthetic and works on the normal RGB system added to pre-defined offsets.

HOSTING THE SOLUTION

One can begin hosting the solution on their computer by running the supplied server.py file with python 3.0 or greater installed on your PC. By default the server will be hosted on port 8080.

HARDWARE REQUIREMENTS

This program will be able to run on any machine capable of supporting a modern browser such as Chrome or Firefox. Memory used will scale directly with the complexity of the code that the user enters into the system.

CONTACT DETAILS

EMAIL: boxedincoding@gmail.com