

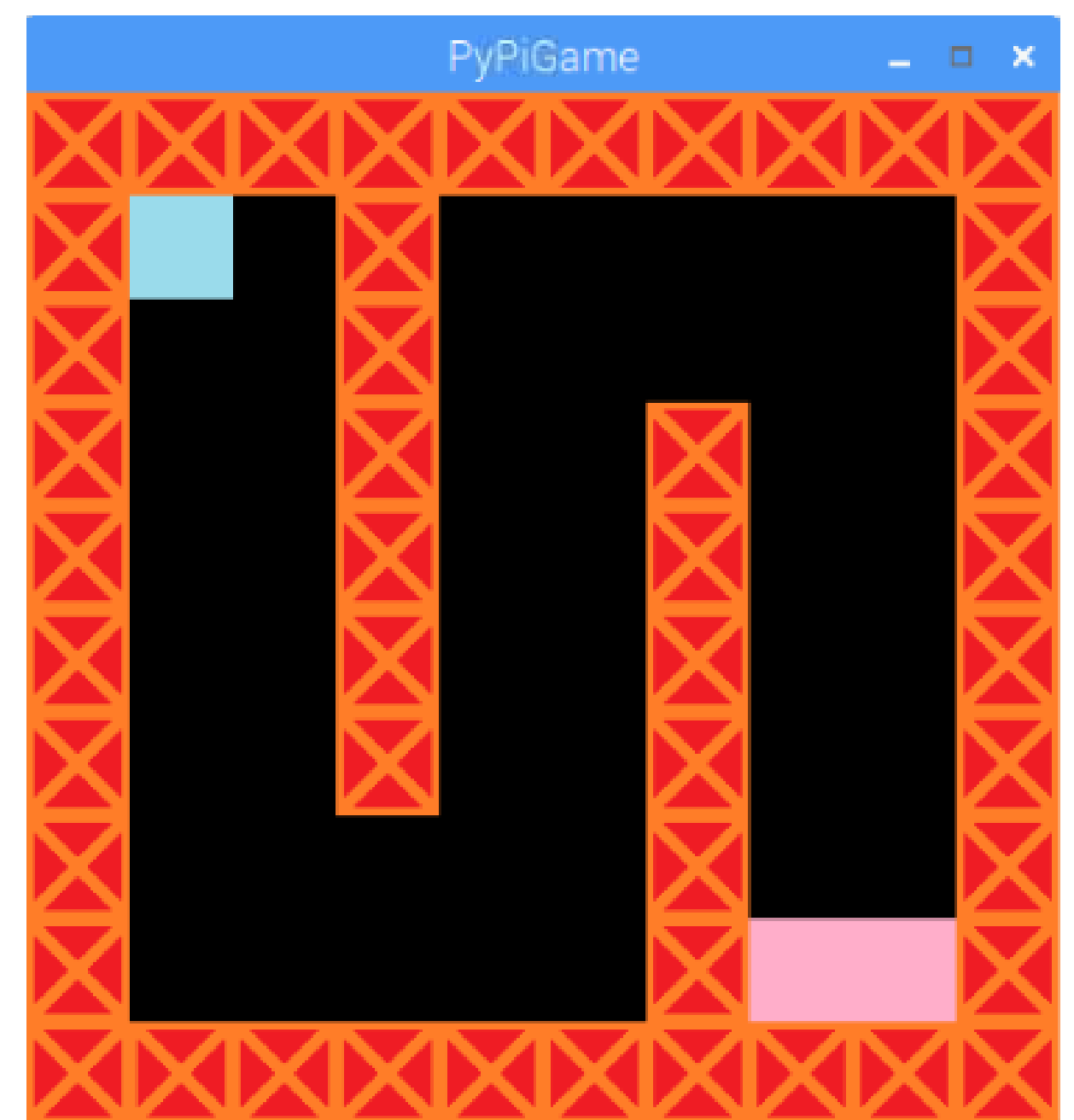
PYPIGAME

Overview

- Programming is a challenging concept to be learnt and high failure rates are common in introduction to programming courses, but when assignments are made in a more engaging way, students are able to learn programming much more effectively.

Aim

- To create a 2D game assignment which students will complete by adding in code specified by the assignment document and can be used in introduction to programming courses to help students learn programming more efficiently.



An example of the PyPiGame assignment Running

Framework

3.1 In the **Maze class**, scroll right to the bottom. You will see the **count_walls_for()** method (**count_walls_for(self): ...**) In this method, use a for loop to count the number of walls on the map and display it to the console.

```
###
def count_walls_for(self):
    # TODO: QUESTION 3.1
    map_arr = self.get_maze(self.maze_choice)
    num_walls = 0

    print("The number of walls counted using the for loop is: ", num_walls)
###
```

```
def count_walls_for(self):
    map_arr = self.get_maze(self.maze_choice)
    num_walls = 0
    for i in range(0, 100):
        if map_arr[i] == 1:
            num_walls+=1
    print("The number of walls counted using the for loop is: ", num_walls)
```

- Assignment document containing the questions for the assignment.

- Code scaffold version of the assignment.

- Assignment memorandum.

Conclusions:

- The questions in the assignment are clear enough for first year computer science students to understand.
- The difficulty of the questions range from easy to challenging,
- The assignment is of a standard which can be incorporated into a first year programming course.



Raspberry Pi with the Sense HAT add-on board



University of Cape Town
Department of Computer Science
Private Bag X3
Rondebosch
7701

Researcher:

Martin Flanagan
FLNMAR011@myuct.ac.za

Supervisor:

Gary Stewart
gstewart@cs.uct.ac.za