# Homework Assignment 7: SuperTurtle – Going Through a Maze

|  |  |
| --- | --- |
| Assigned date | 2024-11-15 |
| Due date | 2024-11-29 |
| Estimate required time | 1 day |

This is an individual assignment.

* You may consult with professor and TA about any aspect of the assignment.
* You may consult with other students only in a general way, e.g., about debugging or Python issues, or questions about wording on the assignment.
* You cannot actively work with someone unless the assignment specifically grants permission to work together with another student.

Remember to follow the CMPT140/166 Coding Standard.

## Purpose

This assignment gives you opportunity to experience object-oriented programming.

## Specification

You have seen the program of using turtle to draw a maze (both in class lecture and in previous assignment). In this assignment you will be “guiding” a turtle to go through a maze.

Specifically, you will finish implementing a class “SuperTurtle”. This SuperTurtle should have the ability to read a list of x/y coordinates that defines a route for going through a specific maze. This SuperTurtle should have the ability to memorize this route (list of x/y coordinates) and go through the maze according to the memorized route.

Please download the following files from Moodle and put them all in the same folder:

* draw\_maze.py
  + this file provides functionality to draw a maze
  + NO NEED to modify this file
* maze\_example\_small.csv
  + this file specifies the maze
  + NO NEED to modify this file
* maze\_example\_small\_solution.csv
  + this file specifies the solution (route) to the maze
  + NO NEED to modify this file
* Point.py
  + this file implements a class that represents a point in 2-dimensional space
  + NO NEED to modify this file
* SuperTurtle.py
  + this file implements a class that represents a turtle that can memorize a route and move according to the memorized route.
  + you WILL NEED to modify this file
  + please see comments in this file for instructions
* test\_driver.py
  + this is the file you would pass to the Python interpreter to execute (i.e. the code entry point)
  + this file calls the associated functions to draw a maze, instantiate (create) the SuperTurtle class, ask the SuperTurtle object to read and memorize the route file and ask it to move according to the route file.
  + PLEASE EDIT THIS FILE so that the maze\_fname and maze\_solution\_fname variables point to the correct path to the CSV files you’ve downloaded.

Further details about the SuperTurtle class:

The SuperTurtle class consist of two instance variables:

1. \_tt: the Turtle object i.e. the turtle that would be responsible for drawing on the screen
2. \_route: a list of x/y coordinates (stored as objects of the Point class).

The go() method would set the x/y origin (i.e. set starting position) and simply iterate through all the Point objects stored in \_route and move the turtle to the absolute x/y coordinates as specified by the Point objects (see the methods getAbsoluteX() and getAbsoluteY())

After you’ve completed the SuperTurtle.py implementation, when you run test\_driver.py, you should see the following maze with the turtle successfully finished the maze.

A screenshot of a computer

Description automatically generated

## Deliverables

You only need to submit your completed implementation of SuperTurtle.py. Please name this file as follows: hw7-LastnameFirstname.py (e.g. hw1-LeungSamuel.py) where Lastname and Firstname are student’s last and first name respectively.

Please submit your source code file hw7-LastnameFirstname.py at <http://learn.twu.ca>

Grading scheme for codes:

* Correct execution (80%)
* Documentation, correct style and structure (20%)

## Hints

After you downloaded all the Python and CSV files, make the appropriate edit to test\_driver.py so that the variables maze\_fname and maze\_solution\_fname point to the correct locations of the CSV files you’ve downloaded. Then, make sure you can execute test\_driver.py without errors. When successfully executed, you should see a maze with the turtle at the beginning (i.e. lower left corner) of the maze.

Correct implementation should take <20 lines of codes.