COSC 1P02 Assignment 3

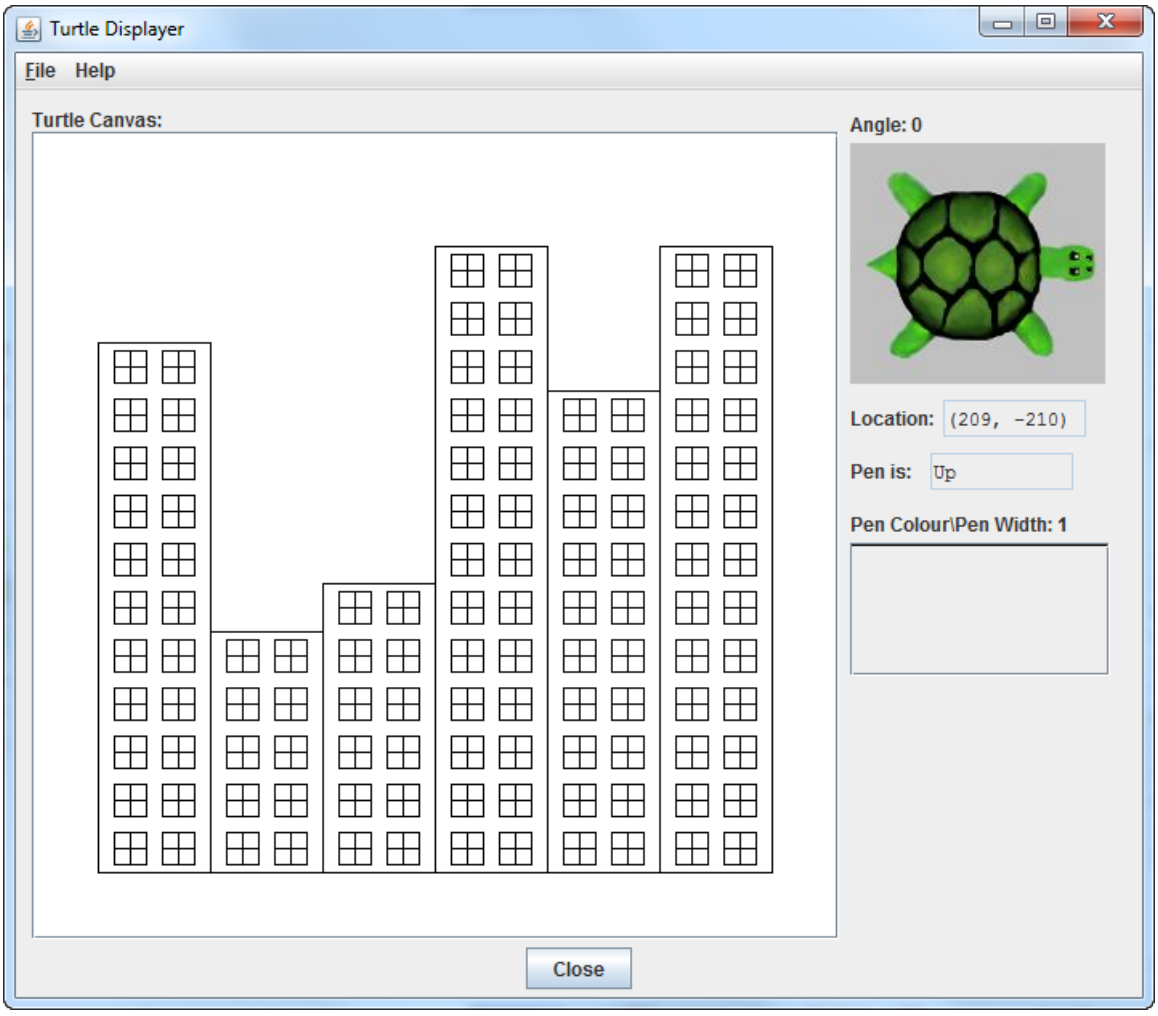
"Method in our Madness"

***Due: Oct. 19, 2018 @ 4:00 pm** (late date Oct. 22 @ 4:00 pm)*

The emphasis for this assignment is methods with parameters. In preparation for this assignment, create a folder called `Assign_3` for the DrJava projects for the assignment.

A Cityscape

Write a Java program to draw a cityscape as seen below. The city consists of some number of buildings. Each building has multiple stories. Each story has windows. Your program should draw a random cityscape as defined below using Turtle Graphics and methods with parameters.



The problem can be defined as such:

- A city consists of 3 to 6 buildings (inclusive), randomly chosen. The cityscape should be centered left to right on the display and ground level should be such that a 15-story building is centered top to bottom.
- The buildings are of width 70 and should be horizontally centred left to right in the display
- Each building is a rectangle with 5 to 15 stories (inclusive), randomly chosen.
- Each story has two windows and is of height 30.

- Each window is four 10x10 squares (remember, squares are also rectangles) centered within the story.

Hints:

- Use a `TurtleDisplayer` with a 500x500 canvas:
  ```
  display = new TurtleDisplayer(yertle,500,500);
  ```
  which creates the display and places `yertle` on the display.
- Use a `FAST Turtle`
- Consider using methods to draw the city of some number of buildings, draw a building of some number of stories, draw a window and draw a rectangle each with appropriate parameters
- Build your program bottom-up, that is, initially simply draw a rectangle, then draw a window, then a building and finally a city.

Submission:

Details regarding preparation and submission of assignments in COSC 1P02 are found on the COSC 1P02 Sakai Site as `Assignment Guidelines` under `Course Documents`. This document includes a discussion of assignment preparation, programming standards, evaluation criteria and academic conduct (including styles for citation) in addition to the detailed assignment submission process copied below.

To prepare and submit the assignment electronically, follow the procedure below:

1. Ensure your folder (`Assign_3`) contains the DrJava project for the assignment.
2. Using DrJava, print (as a pdf file, e.g. using "printer" `Microsoft Print to PDF` or similar) the `.java` file for your assignment using the name *`ClassName`*`.pdf` where *`ClassName`* is the class name (i.e. same name as the `.java` file) and save the `.pdf` file at the **top level** of the assignment folder (i.e. directly within `Assign_3`).
3. Run the program. When the display is finished (i.e. `Close` button visible), select `Print Image of Window...` from the `File` menu on the `TurtleDisplayer` and direct the output to `Microsoft Print to PDF` saving the `.pdf` file at the **top level** of the assignment folder (i.e. directly within `Assign_3`) using an appropriate name (e.g. `Output.pdf`).
4. Create a `.zip` file of your submission by right-clicking on the top level folder (i.e. `Assign_3`) and selecting `Send to/Compressed (zipped) folder`. A zipped version of the folder will be created. Use the default name (`Assign_3.zip`).
5. Log on to Sakai and select the COSC 1P02 site.
6. On the `Assignments` page select `Assignment 3`. Attach your `.zip` file (e.g. `Assign_3.zip`) to the assignment submission (use the `Add/Remove Attachments` button and select `Browse`). Navigate to where you stored your assignment and select the `.zip` file (`Assign_3.zip`). The file will be added to your submission. Be sure to check the `Honor Pledge` checkbox. Press `Submit` to submit the assignment. You should receive a confirmation email.

DrJava

The `.zip` folder you submit should contain the project folders for the two parts, including all files relevant to the project—the `.java` and `.class` files for the assignment—and the `.pdf` files for program listings and output at the top level.

Other Platforms

If you are using an IDE other than DrJava to prepare your assignment, you must include the `.java` source files and the `.pdf` files described above for each part as well as an executable file (likely `.class` or `.jar`) that will execute on the lab machines.