CMEE Masters: Computing Coursework Assessment

Assignment Objectives: To work on a series of computing/programming exercises and problems in a coherent, modular, reproducible workflow under version control.

Note that:

- *The overall assessment will typically have significantly lesser marks than a simple weighted average of each week's points because the overall assessment is based on not just the "Computing Coursework Assessment Criteria", but also the the "Marking Criteria for Exams, Essays and Coursework". Both sets of marking criteria are in the Assessment Appendix of the online TheMulQuaBio notes and git repository.*
- *In your 1:1 post-assessment feedback session, we will discuss where you gained or lost marks, and what you could have improved further. To the extent possible, please come with questions about specific scripts based upon the overall and weekly feedback you have received. This may require you to compare your code with the solution code in many cases.*

Student's Name: Kayleigh Greenwood

1 Specific feedback

1.1 The Good (what you did well!)

1. Found all the core CMEE weekly directories in your parent directory.
2. Your code and repo directories are generally cleanly and logically structured.
3. Your Git repo size when I checked week 7 was about 4 MB – nicely compact! This suggests you correctly suppressed unnecessary files from version control, and did not commit excessively. It could also mean that you did not commit enough, and/or somehow along the the way lost parts of your git history – but we don't check these possibilities!
4. You had an overall readme file, as well as one within each week. The Readmes were clear, and comprehensive, including language version numbers. Good work!
5. Your Python is generally very nicely modularised – this is excellent as it is a very Pythonic way of doing things. Good job also remembering all the docstrings.
6. Very good job with the coding overall. Good attention to detail, with only a couple of IndentationErrors (see below), and generally very conscientious handing of potential errors like missing command line arguments or data (with the one exception noted below).
7. Your Groupwork practicals were all in order, and your group did well in collaborating on it. More feedback on this in the 1:1 sessions.

1.2 The Bad (errors, missing files, etc)

1. `get_TreeHeight.py` throws an IndexError if no command line argument is supplied. This could have been caught and treated with a more informative error message.

2. `LV1.py` and `LV2.py` throw IndentationErrors due to the extra indents (e.g. after the "# # figure 1" or "# # figure 2" comments). Recall that in Python, indentation is not merely an aesthetic choice, but actually determined which lines of code "belong" to, say, a loop or a function.

1.3 The Ugly (niggling issues like commenting, cosmetics, complexity of code, etc)

1. In your readmes you included some note to the language and version numbers, but could stand to include dependencies (e.g. packages used) as well. Also check out this resource: `https://github.com/jehna/readme-best-practices`. As you become a seasoned programmer, you will learn to make the readme file descriptions even more informative yet succinct.

2. You had a .gitignore to control which files were under version control, which is good, though you might also have opted to make week-specific exclusions. You will likely find this useful: `https://www.gitignore.io`.

3. You comment your code very carefully, indeed you sometimes go a little overboard and are too verbose, or comment very basic commands that don't really need it. This is probably better than not commenting enough (or at all), so don't worry too much, just be aware that comments are most useful when used to pre-emptively explain somewhat complex logic or a somewhat unusual/counter-intuitive/non-obvious way of achieving something. This will improve with experience, as you will begin to get a feel of what is "common-knowledge" among programmers, and what stylistic idioms are your own and require explanation.

2 Overall Assessment

Overall an excellent job. Nicely modularised and generally well structured code with just some careless indentation errors causing problems. Very well done!

Provisional Mark: 73%

Signed: Alexander Kier Christensen & Samraat Pawar

March 23, 2022