Marking Notes for NWEN241 Assignment 1:

Setup:

- Download the marking code from https://gitlab.ecs.vuw.ac.nz/jpfender/nwen241 or from the NWEN241 staff email.

- If you're using GCC instead of clang (e.g. using cygwin), change the flag in MAKEFILE of the marking code from (`CC = clang`) to (`CC = gcc`).

- To test a students code, simply copy their `editor.c` file into the marking code before compiling.

Compilation and Execution :

- To compile the marking code, run `make`. This will create more than one executable.

- When substituting in another `editor.c` file, run `make clean` first, to ensure it compiles the new source file with `make`.

- If `make` fails to compile the project because the students haven't implemented/completed one or more tasks, you can compile tests for each question separately using `make tX` where `X` is either 4, 5, 6, 7 for the given assignment tasks.

- To run all tests, use `./a1test`, but to run each test individually, run the corresponding executable (e.g. `./t4`)

- **Take all test failures with a grain of salt.** Tests are not distributed to test conventional use cases, but to expose edge cases.This means a student's code may be very close to the correct solution, but may still fail most or all of the tests. However, they still deserve marks based on how close they are to the solution.

Name:	Marks:	Notes:
Part 1: (Task 1, 2, 3)	**33**	Partial marks for incorrect answers but correct idea/working/explanation

Part 2: Compilation	7	**[10]** compiles without any warnings **[8-9]** minor warnings **[5-7]** quick fix compiler errors (couple minutes) **[1-4]** serious compiler errors (up to 5 minutes) **[0]** can not be fixed within reasonable time
Part 2: Comments	7	**[10]** Descriptive, concise and appropriate comments **[8-9]** Relevant comments **[6-7]** Few or non-descriptive comments **[0-5]** No comments, depends on readability of code
Part 2: Coding Style	7	**[10]** Completely consistent and clean. **[7-9]** Occasionally inconsistent or unclean **[4-6]** Little to no effort made towards coding style **[0-3]** Intentionally obfuscated
Task 4: Correctness	14	**[14]** No test failures **[11-13]** Fails some but not all test cases. Minor fixes produce no test failures. **[7-10]** Fails most/all test cases. Minor fixes produce few or no test failures. **[1-6]** Fails most/all test cases. Clear effort made, but is difficult to fix. **[0]** Incomplete or no code provided
Task 5: Correctness	17	**[17]** No test failures **[13-16]** Fails some but not all test cases. Minor fixes produce no test failures. **[8-12]** Fails most/all test cases. Minor fixes produce few or no test failures. **[1-7]** Fails most/all test cases. Clear effort made, but is difficult to fix. **[0]** Incomplete or no code provided
Task 6: Correctness	8	**[8]** No test failures **[7-8]** Fails some but not all test cases. Minor fixes produce no test failures. **[4-6]** Fails most/all test cases. Minor fixes produce few or no test failures. **[1-3]** Fails most/all test cases. Clear effort made, but is difficult to fix. **[0]** Incomplete or no code provided
Task 7: Correctness	7	**[7]** No test failures **[5-6]** Fails some but not all test cases. Minor fixes produce no test failures. **[3-4]** Fails most/all test cases. Minor fixes produce few or no test failures. **[1-2]** Fails most/all test cases. Clear effort made, but is difficult to fix. **[0]** Incomplete or no code provided