

CHECKLIST TO GO THROUGH BEFORE ASSIGNMENT SUBMISSION

COMP9021 PRINCIPLES OF PROGRAMMING

- ☐ The name of the file that contains the `main()` function is correct.
- ☐ The program compiles and runs on a School machine.
- ☐ The program is stylistically compliant with our norms, and in particular, is correctly indented. This is supported by the fact that the `mcstyle` script has been run (recall that `mcstyle` needs the `style_sheet.txt` file whose default customisation might have to be changed, depending on preferences) without complaining that any fixed constraint or any other stylistic mismatch has been detected.
- ☐ The program does not contain excessive indentation levels. The specifications mention the maximum number of indentation levels accepted; this value has been assigned to `Maximum level of indentation` in `style_sheet.txt`, and the `mcstyle` script has been run without complaining that any statement with excessive indentation depth has been detected.
- ☐ The program has been tested not only with respect to the examples provided with the assignment specifications, but also on a well designed suite of tests that encompass “limiting” as well as “typical” cases.
- ☐ The output of the program is very precisely what it is meant to be as described in the assignment specifications. The program outputs no blank line or extra message meant to make the output “look better”. The text that is output is correct and has no typo. Confidence is great that the program is free from mistakes of this kind because the output of the program (redirected to a file in case it is not meant to be saved in a file but sent to standard output) has been compared with the expected output for the examples provided using `diff`, that exits silently when both files are identical.
- ☐ The program does not read from a file an input that is intended to be entered from the keyboard, even when that input is likely to be stored in a file and redirected to standard input for practical purposes. The program does not read from standard input any data intended to be provided as command-line argument.
- ☐ The program never sends any message to standard error. Indeed, the output is meant to be either sent to the screen or saved in a file; the program does not save in a file any output intended to be sent to the screen even when that output is likely to be redirected to a file for practical purposes. When the output of the program is intended to be saved into a file and the specifications impose a particular name—that might be derived from the name of an input file—, then that name is correct.
- ☐ The code contains no magic number. Macros have been used to define the specific values of some parameters (such as the maximum value that a number is expected to take). The value of characters is referred to by using single quotes (*e.g.*, the value of `A` is referred to by `'A'` and not by `65`).
- ☐ The code is properly commented, with comments meant to facilitate the task of understanding and modifying the program in a few years time, and with nothing but useful comments that do not tell the obvious.
- ☐ All functions are reasonably short.
- ☐ Functions and variables are given as much as possible evocative names that facilitate the task of understanding the program.
- ☐ Every statement is reasonably short. In particular, no test is a boolean combination of conditions that span over many lines.
- ☐ Great time and effort have been invested into making the underlying logic as simple as possible. When simplifications have been envisioned, they have been implemented, even if substantial parts of the program or even the whole program had to be entirely rewritten.

Submit all files expected to be submitted, including your customised `style_sheet.txt`. Your program might have successfully passed all your tests, and fail some or all of my tests. This might be because I will use different tests to those you have been provided with and to those you have created. This might also be because your program is buggy. A correct program gives correct results on all machines, always. A buggy program can potentially give incorrect results on some machines, sometimes.