Grading Rubrics

[CS 120] High-level Programming I: The C Programming Language

1. Overview

During the semester there will be about 12 laboratory exercises to work on during the laboratory classes, and about 5 programming assignments to work on outside the class. These, collectively referred to in this document as the assignments, constitute 25% of the final grade in this course.

The assignments will use the C programming language. More specifically, all programs must adhere to the C90 standard, which is what this course is about. Every assignment will consist of a program specification, which describes the functionality that needs to be implemented, and it may include a list of functions that you must implement to complete the assignment. You must strive to follow all of the directions exactly as specified in the specification.

Your submissions will be evaluated with a modern Microsoft Visual C++ compiler (Visual Studio 2017 or newer) and GNU gcc compiler (7.4.0 or newer). These compilers are installed on DigiPen Singapore computers. You must strive to follow all of the directions exactly as listed in the specification.

2. Rubrics

Assignments are graded on the scale 0 to 100 marks. Submissions must be uploaded to the DigiPen Distance Learning course management system (*Moodle*), and they are assessed in the following way:

- Critical penalties
  - **100 mark penalty** will be imposed for plagiarism. Additional consequences will follow.
  - **100 mark penalty** will be imposed for any submission deemed late by *Moodle*. Do not wait until the last minute, as uploading the work after the deadline may be impossible.
  - **100 mark penalty** will be imposed for any submission with a missing required file, including the *check list*, with files with incorrect names or formats, or with extra files, such as binary output of the compilation process. Follow the specification document.
  - **100 mark penalty** will be imposed for any submission that does not build properly due to errors – where at least one driver file does not compile (or does compile when it should not) with required flags (except for the flag for treating warnings as errors) with at least one compiler (gcc and VC++). Test your code.
- Major penalties
  - **50 mark penalty** will be imposed for each essential memory leak.
  - **50 mark penalty** will be imposed for crash using the test data provided with the assignment. You have a chance to test your code in advance.
  - **25 mark penalty** will be imposed for each mismatched output per driver file, per input file (where provided). You have a chance to test your code in advance.
- Minor penalties
  - *10 mark* penalty will be imposed for violations of the specification for each instance.
  - *10 mark* penalty per warning will be imposed for any submission that does not build properly due to warnings – where at least one driver file does not compile (or does compile when it should not) with required flags (including for the flag for treating warnings as errors) with at least one compiler (gcc and VC++).

- *10 mark* penalty will be imposed for each excessive character in lines longer than 80 characters (including whitespaces; tabs are counted as 4 spaces).
- *10 mark* penalty will be imposed for each excessive line in functions longer than 50 lines (including curly brackets, but excluding empty lines and comments).
- Up to *5 mark* penalty will be imposed for each violation of general conventions and style guidelines (for instance, inconsistent indentation, varying bracket placement, or erratic naming conventions).

The list above is non-exhaustive. The lecturer reserves the right to impose reasonable penalties for the code that violates general practices or does not match the specification in an obvious way that has not been mentioned above. In exceptional cases, the lecturer reserves a discretionary right to allow resubmission or submission after the deadline.

3. Assignment Submission

When submitting source files written in C, you must adhere to the following guidelines. All files must be in a single ZIP archive (even if you are submitting a single file). The name of the ZIP file must follow this naming convention:

```
<login>_<class_name>_<assignment number>.zip
```

For example, if a student with login name *foo.bar* submits assignment *#1* for course *CS 120*, the appropriate file name would be *foo.bar_cs120_1.zip*. Note that the file names are case-sensitive and must be in lowercase, exactly as described above. The ZIP archive **should not contain a single root folder** with individual files and subfolders inside, but it should rather contain such files and subfolders directly in the root location.

Detailed instructions may be provided with each assignment specification. Every assignment submitted on *Moodle* contains a date and time stamp. There is more than enough time in your schedules to complete assignments on time. Of course, if you wait until the deadline is near to begin your assignment, you may not be able to complete it on time. Time management is your responsibility.

4. Late Submission

Each assignment provided will be accompanied with a due date and time (typically, midnight), which will be clearly stated on the assignment submission page. A 100% penalty will be imposed on any submission deemed late by the DigiPen Distance Learning course management system (*Moodle*).

Students, who want to request for extensions, should provide valid reasons to justify their case. This will be handled on a case-by-case basis by the lecturer. Request for extensions after the deadline will not be accepted except in extraordinary cases.

5. Grade Appeal

If a student is dissatisfied with the grades, they should firstly raise up the matter to the instructor. As the submitted work may be assessed by grading assistants, this will give the instructor a chance to review the submission in person. If the student is still not satisfied, they can do a formal grade appeal through the Registrar's Office. A formal committee will be assembled to resolve the appeal. In such cases, students must fill-in the Grade Appeal Form available at Front Desk and submit it to the Registrar's Office, along with supporting documents.

6. Academic Integrity

DigiPen Institute of Technology Singapore stands for academic honesty, and professional integrity. As this course requires students to submit work for assessment, through this policy DigiPen would like to highlight the importance of the proper moral conduct and ethics.

Academic dishonesty in any form will not be tolerated in this course. Cheating, copying, plagiarizing, or any other form of academic dishonesty (including doing someone else's individual assignments) will result in, at the extreme minimum, a zero on the assignment in question, and could result in a failing grade in the entire course or even expulsion from DigiPen.

It is permissible to discuss assignments, but not solutions, with other students in the class; the solutions must be recognizably your own. With the internet as a readily accessible source of information and help, students may feel that plagiarism is ambiguous, and thus be unable to determine what it constitutes. Here are some general guidelines to help make the distinction:

- Do **NOT** copy-paste any works online. Using works that are not yours is plagiarism.
- Do **NOT** ask online communities (such as stack overflow, unity forums etc.) to solve your bugs and code issues by providing your code segments. Asking others to solve your issues is work not done by you, and thus it is plagiarism.
- You may learn from sources online, understand the workings and concepts, and implementing them again via **your own efforts**. A good habit is to assume that you will be tested on the things you learn online, and if you are unable to answer the questions then you should not use said works.
- You may ask online communities about general problems, and use their insights **to work on your problem**.
- These applies to all sources on any medium, be it the internet, textbooks, friends or social media. It is the content that is important, not the medium they are on.

The bottom line test is to ask yourself "**Did I work on this?**" If you did not, then you should not use it. Learn from it and work it out yourself.