CS251 - Project 1: C++ Programming Example

Out: January 15, 2018 @ 9:00 am
Due: January 22, 2018 @ 9:00 am

1. Overview

Write a simple C++ program that reads in an example file and outputs an example file. The project also serves to familiarize you with the programming environment you will use during the semester.

2. Detailed Description

You are going to write a program that reads text and output it in multiple column format.
The input consists of a first line containing a single number C, the number of columns in the subsequent lines. The second and later lines contain C integers per line separated by whitespace. All numbers will be greater than or equal to 0 and less than 1000. Your program should output the data organized into a columns as follows:

- Each number occupies the space of 3 digits, right justified, and with no leading zeros
- Columns are separated by the string: " | " (space, pipe symbol, space)
- The first and the last line contain exactly 6C-3 dash characters ("-"), i.e. the same number of characters as each of the remaining lines
- There is no other extraneous whitespace or trailing separators

For example, given a file with the following lines:

```
3
1 12 103
104 15 6
```

The output file should be:

```
--------------------
  1 |  12 | 103
--------------------
104 |  15 |   6
--------------------
```

3. Grading and Programming Environment

Assignments will be tested in a Linux environment. You will be able to work on the assignments using the Linux workstations in HAAS and LAWSON (use your username and password). Compilation will be done using g++ and makefiles. You must submit all the source code as well as the Makefile that compiles your provided source code into an executable named "program".

Your project must compile using the standard g++ compiler (v 4.9.2) on data.cs.purdue.edu. For convenience, you are provided with a template Makefile and C++ source file. Note some latest features from C++14 are not available in g++ 4.9.

The grading process consists of:

1. Compiling and building your program using your supplied makefile.
2. The name of produced executable program must be "program" (must be lowercase)
3. Running your program automatically with several test input files we have pre-made according to the strict input file format of the project and verifying correct output files thus follow the above instructions for "stdout" output precisely – **do not "embellish" the output with additional characters or formatting** – if your program produces different output such as extra prompt and space, points will be deducted. (It is fine however if you print your extra output to "stderr" if it helps in debugging.)
4. Inspecting your source code.

Input to the programming projects will be via "stdin" and output must be to "stdout". For example, in command line form, your program will be run as:

program < input-test1.txt > output-test1.txt

The file output-test1.txt will be tested for proper output.

Important:
1. If your program does not compile, your maximum grade will be 25% of the grade (e.g., 2.5 out of 10) -- no exceptions.
2. Plagiarism and any other form of academic dishonesty will be graded with 0 points as definitive score for the project and will be reported to the corresponding office.

Submit Instructions:
The programming assignment must be turned in by the due date and time using the turnin command. Follow the next steps:

1. Login to data.cs.purdue.edu (you can use the labs or a ssh remote connection).
2. Make a directory named <yourName_yourSurname> and copy all the files needed to compile and run your code there (i.e, **makefile, all your source code files including headers and any other required additional file**).
3. While in the upper level directory (if the files are in /homes/jsmith/john_smith, go to /homes/jsmith), execute the following turnin command:
     turnin -c cs251 -p project1 your_folder_name
   (e.g. turnin -c cs251 -p project1 john_smith)
   (**Important:** previous submissions are overwritten with the new ones. Your last submission will be the official and therefore graded).
4. Verify what you have turned in by typing
     turnin -v -c cs251 -p project1
   (**Important:** Do not forget the **-v** flag, otherwise your submission would be replaced with an empty one).