CMP_SC 3050 Homework 1
Due: 11:59:59 pm, 01/29/2015

The first homework for CMP SC 3050 is a ***programming assignment***. The first part of the homework specifies the exact problems that your submission should solve and the second part describes the constraints that you **must** follow (no exceptions). Absolutely no late work will be accepted.

Specification:

For this programming assignment, one compressed file containing the directory of all of the assignment files should be submitted to Blackboard prior to the due date. Please do **not** submit an executable. Your assignment must compile on Babbage in order for you to get any credit.

The assignment should:

- Read in a file representing a list of numbers. The numbers can be positive or negative. A positive number is represented as a string of digits and a negative number is represented as a minus sign (-) followed immediately by a string of digits. Numbers shall be separated by whitespaces, newlines and tab-spaces. A sample input file can be found on Blackboard.

- Write procedure(s) to read and display the number of integers input.

- Write procedure(s) to compute and print the maximum number in the list.

- Write procedure(s) to compute and display the median of all the numbers. Recall that the median of a list of n numbers is defined as follows

    - If n is odd, then the median is the number such that there are exactly (n-1)/2 less than the median in the list.
    - If n is even, then the median is the average of the (n/2) th smallest and the (n/2+1)th smallest numbers in the list.

Constraints:

The following is a list of constraints for this assignment; failure to adhere to any of these constrains will result in a loss of points or even a zero.

- The assignment must be built using a *makefile*. If you are not familiar with makefiles, contact a TA or come to office hours. The makefile **must** have at least 3 targets: *maximum*, *median* and *clean*.
    1. The target *maximum* should, given an input file storing a list of numbers, compute the maximum and output the number of elements in the list as well as their maximum.
    2. The target *median* should, given an input file storing a list of numbers, compute the median and output the number of elements in the list as well as their median
    3. The target *clean* should remove all object files and executables from the directory.

- The procedures for the four requirements (reading the input, counting the number of elements in the list, computing their maximum and computing their median) **must** be written in different files.

- The assignment can be completed in either C or C++. No other programming language is required.

- Built-in data structures and external libraries may not be used for this assignment. If the program requires a stack, linked list, or any other structure, it is up to you to provide it. If you have any doubt on whether or not any technique you wish to use is acceptable, do not hesitate to ask.

- A moderate amount of error checking and resource management is required. Even if you do some error recovery, you must report errors in the input. Your application should ensure that each line from the input file is properly formatted, that the file is successfully opened, the file is successfully closed upon reading of the file and that all allocated space is de-allocated at the exit.

A moderate amount of formatting and documentation is required. Comments should be descriptive and used to illustrate the purpose and inner workings of an algorithm or function; they should not be used to annotate each line or self-evident logic.

Grading:

There are 11 points possible for this assignment. The grade breakdown is as follows:

- 4 points for correct design of Makefiles.
- 2 points for error checking and resource management.
- 1 points for general programming style and adherence to the constraints.
- 1 points for correctly inputting the list of numbers and counting the number of elements in the list.
- 1 point for computing the maximum.
- 2 points for correctly computing the median.

If the program fails to compile or crashes due to a runtime exception, a grade of zero will be assigned.