

CST8134 Assignment 1: proper

You are about to be assigned to a major project by your supervisor that you've been told will involve embedded processors. You decide that you need to brush up your skills at Motorola 68000 assembler language in preparation.

There's not much available as a development environment on the standard server, so you decide just to read a line of data from the keyboard, proper-case it and print it out, then read another until you receive an end-of file condition.

In proper-cased text, the personal pronoun "I" must be upper case as must the letter beginning the first word of a sentence (the first word of the first line, as well as the first word following any of (but not the quotes!) ".?!"). Everything else will be lower case.

Use your own small tests during the development, but when you consider your program to be ready, test it with the following four lines of input (the text is freely adapted from "Through the Looking Glass" by Lewis Carroll, during the portion where Alice meets and talks to Humpty Dumpty):

"when i use a word," he said in RAtHER a scornful tone, "it means just What I chOose it to mean--neither MORE nor LESS!" "The question is," she argUed, 'whether you CAN make words mean so many diffERent things?' "the question Is," he retorted, "which iS to be MAstER--that's all."

You will hand in your assignment at your lab start in 3 weeks time (week 7). No disk, email, or Blackboard submission is required: use paper. This must include:

- The standard cover page, with a table of contents;
- A Design Document (write it in MS Word or OpenOffice) containing:
 - The problem statement in your own words, and some observations on requirements that may be implied or unstated;
 - The PDL for the program (do not also include it in your source code);
 - A data dictionary in table form that lists the registers you expect to use, as well as each item in the DATA and TEXT 1 sections (the name, type, size in bytes, and a brief purpose for each);
 - A test plan to confirm that all aspects of your program work correctly, and that incorrect input will be dealt with reasonably (use a table to show the test number, the input to use, the expected result, and the purpose of the test);
- A cleanly printed copy of your programs .lis file, including reasonable comments (stating why, not what), both in-line and block (each function including your main program must have a block comment header at its start much like a C++ function);
- Your printed test results, consisting of:
 - an updated test plan showing the actual results of the tests (briefly);
 - a screen-shot showing the results for the required data given above.

The program need not be complex. You will note that the bdb emulation environment includes ToUpper and ToLower library calls (but you don't have to use them), and that you've already built a line reader (lab 4, Read a line) that can be re-used for this program.