Word Count

CISC 221 – Assignment 1

Description

Write a program in C that reads from the standard input, i.e. keyboard, and counts the number of lines, words, and characters up to the EOF. **It should mimic the built-in Unix/Linux command of `wc`.**

- It will only need to work with texts encoded with ASCII.
- When reading from the input, before reaching EOF, the texts are assumed to be divided into lines, marked by `'\n'`, the newline character.
- To simplify the case, we need to consider the following delimiters between words
  - newline `'\n'`
  - tab `'\t'`
  - space `' '`
- The rest of the texts are grouped as "words" for counting.
- You are limited to using `<stdio.h>` and `<string.h>`.

Take the following steps (I have also uploaded a tutorial video on onQ):

1. Activate your CASLab account: https://courses.caslab.queensu.ca/accounts-management/
2. Remote log into one of the CASLAB Linux machines (e.g. linux.caslab1.queensu.ca) through your *Visual Studio Code* and setup an `SSH` connection through the Remote Explorer by following the instructions in (Link).
3. Create necessary directories to best organize your work. You can create the directories visa the command prompt in the terminal in Visual Studio Code
4. Create a new text file via the Visual Studio Code to write your C source file. Save it as `wc.c`.
5. Ensure that the Linux prompt in the terminal is on the same directory where the file is saved. You may need to change directory via the `cd` command.
6. Compile it with `gcc -o wc wc.c` so that the output executable is `wc`.
7. Run your program by issuing `./wc`. In the terminal, type in arbitrary lines of texts, and end it with Ctrl-d. If you have a text file handy, say `ABC.txt`, you can also feed it to your program using I/O redirection to save yourself from repetitive typing, that is, `./wc < ABC.txt`.
8. Make sure your output agrees with the built-in command of `wc` by checking it with `wc < ABC.txt`.

Deliverables

- Submit the single file `wc.c` via OnQ.
    - Ensure that your source code is well-documented and readable.
    - Make sure it is tested on the CASLab machines.

Hints

Your wc.c program should mimic the Linux build-in word count. Which means that punctuation marks should be counted if stand-alone (e.g., h . h are three words). However, they should not be counted if they are appended to a word (e.g., h, h are two words). Punctuation marks should be counted as characters in both cases. Spaces and new lines (\n) should not be counted as words but counted as characters.

Steps to create text file on Linux and use the built-in wc command.
You can show the file by Linux> cat filename.txt

Creating the file:
Linux> cat > filename.txt
Provide your input and then press `Ctrl` + `D`

You can show the file by the following command
Linux> cat filename.txt

Use the Linux built-in wc by the following command.
Linux> wc filename.txt