

# Word Count

CISC 221 – Assignment 1  
Due: January 20, 2022, 11:30pm

## 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 `' '`
  - period `'.'`
  - semicolon `';'`
  - colon `':'`
- 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:

1. Activate your CASLab account: <https://courses.caslab.queensu.ca/accounts-management/>
2. Remote log into one of the Linux machines (e.g. `linux.caslab.queensu.ca`) using an SSH client (Windows) or the `ssh` command (Unix/Linux/macOS).
3. Create necessary directories to best organize your work.
4. Use a text editor (e.g. `nano`, `vi`, or `emacs`) to write your C source file, call it `wc.c`.
5. Compile it with `gcc -o wc wc.c` so that the output executable is `wc`.
6. 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 `LoremIpsum.txt`, you can also feed it to your program using I/O redirection to save yourself from repetitive typing, that is, `./wc < LoremIpsum.txt`.
7. Make sure your output agrees with the built-in command of `wc` by checking it with `wc < LoremIpsum.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.