Directions to use word counter program:

These directions are to run the code in Google Collab. However, the code is in C and intended for Unix-like systems so it may be able to run on operating systems like MacOS and Linux distributions. I used Google Collab because it was difficult to work with Windows API on my machine when writing code in C.

Mount Drive:

The program was designed and intended to be used in Google Collab. Since the code is intended to read from files in a directory, you will need to mount your drive so that the code is able to access the directory. The code for that and the resulting output should look as follows:

```
[1]  from google.colab import drive
     drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

You will also have a GUI pop up asking for permissions that must be accepted.

Add C code to code chunk:

Next you should create a code chunk in Google Collab and include this code at the top:

```
%%writefile my_program.c
```

This will allow the backend engine that normally processes Python code to write the C file into the system. After this line of code paste the C code into the same chunk. Then run the code chunk which creates an executable of the program.

Execution Code Chunk:

```
directory_path = '/content/drive/MyDrive/calgary'  # Replace with your actual directory

!gcc my_program.c -o my_program
!chmod +x my_program
!./my_program $directory_path
```

Next create a new code chunk to run the compiled C program. This contains the directory path for the directory which should be stored as a folder in your Google Drive. Then you must edit the path to reflect the path of the directory in your drive. This then runs the code while giving access to the directory with the files to analyze. After running this code chunk you should get an output that reflects the following image:

```
Single-threaded - CPU time: 0.000142 seconds, Memory usage: 127412 KB
Top 50 words in file: /content/drive/MyDrive/calgary/bib
%a: 1195
%t: 725
%d: 699
%j: 455
%p: 443
%k: 432
*: 401
of: 372
%o: 370
%v: 304
%i: 263
and: 256
%c: 254
the: 236
%n: 211
a: 194
in: 158
computer: 146
for: 132
1984: 128
proc: 127
1983: 124
%r: 116
systems: 111
on: 107
ieee: 104
1985: 99
university: 88
report: 82
```

The output gives the results of each single-threaded and multi-threaded process along with the runtime and maximum CPU usage.