Robel Ayelew   ID:922419937

Github: RobelKasahun   CSC415 Operating Systems

Assignment 4 – Word Blast

Description:

This assignment is about reading a text file named "WarAndPeace.txt" and using the text data to find ten words that are 6 or more characters long and have the highest frequencies. In other words, the aim of this assignment is to find ten words in the file that occurred or were repeated more times than any other words inside the file. Most importantly, processing or finding the top ten words that have the highest tallies or frequencies must be done using threads, meaning the whole data of "WarAndPeace.txt" will be divided among a given number of threads and each thread will process chunks of data of the file.

Approach:

My approach to this assignment will be to first open the file in a read-only mode because the first and main task is to read or get the data from the file named "WarAndPeace.txt". I will get the whole data from the file and will divide the data among given threads, therefore each thread will process its chunk, meaning each thread will get its chunk of data and it will parse and count the number of frequencies each word has. After all the threads are finished processing, I will join all threads and will get the final result. At this point, all threads will finish processing, therefore, I will sort the results I received from all threads and will print the 10 words that have the highest frequencies.
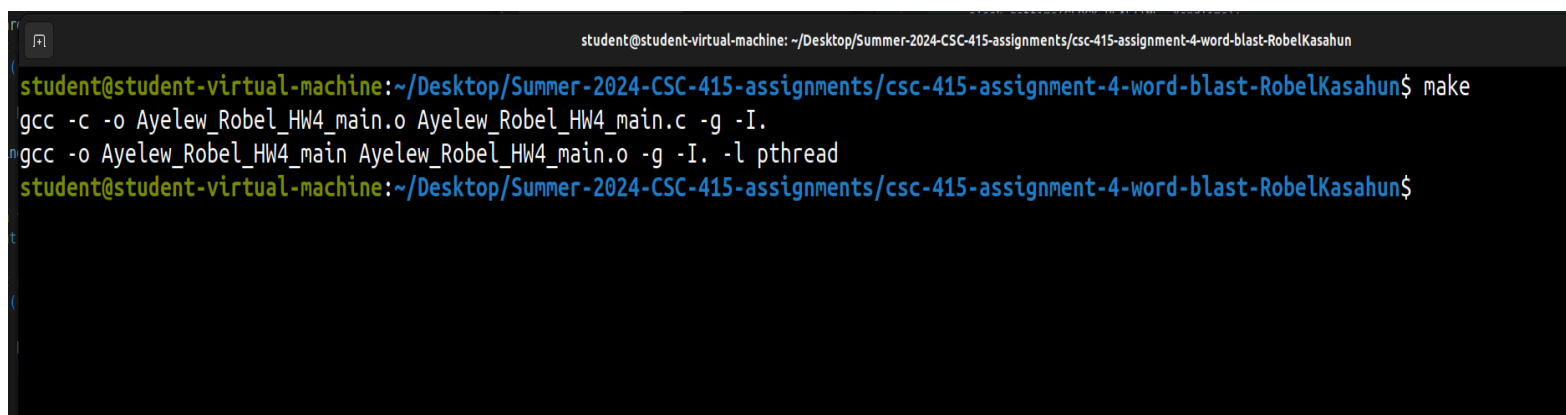
Issues and Resolutions:

I had a lot of issues with this assignment and resolved most of them, but the only issue that persisted until the submission time was that I could not figure out how to the results meaning the top ten words that have the highest frequencies between different threads counts. To elaborate more, I executed the code using make run with thread 1 and the result is identical to the results you provided on the assignments github, but when I execute the source code using threads 2, 4, 8, and 16, the results I got are not identical, therefore I still do have the issue and It is impossible for me at this time to make my solution to consistent results regardless how many threads I use to execute the source code.

Analysis:

My analysis of this program is that when I executed my source code using 1 thread the time it took to execute the code was about **_1.443615596_** seconds, but when I used 2 threads the time it took to execute the code was about **_0.774233189_** which is almost half time compared to the time we got when we used 1 thread. Furthermore running the code using 4 threads took about **_0.741172400_** seconds which is about **_0.03_** fewer seconds compared to the result we got when we run our code using 2 threads. Last but not least, when I executed the code using 8 threads, the time it took to run the code was about **_0.738474722_** seconds which is virtually identical to the time we got from using 4 threads. Finally, these results are based on one-time execution of the code using various threads, therefore running multiple times using different threads will not give you the seconds I provided above.

Screenshot of compilation:

Screenshot(s) of the execution of the program:

```
student@student-virtual-machine:~/Desktop/Summer-2024-CSC-415-assignments/csc-415-assignment-4-word-blast-RobelKasahun$ make run RUNOPTIONS="WarAndPeace.txt 8"
./Ayelew_Robel_HW4_main WarAndPeace.txt 8


Word Frequency Count on WarAndPeace.txt with 8 threads
Printing top 10 words 6 characters or more.
Number 1 is Pierre with a count of 1898
Number 2 is prince with a count of 1844
Number 3 is Natásha with a count of 1182
Number 4 is Andrew with a count of 1026
Number 5 is himself with a count of 1011
Number 6 is Princess with a count of 921
Number 7 is French with a count of 837
Number 8 is before with a count of 801
Number 9 is Rostóv with a count of 769
Number 10 is thought with a count of 760
Total Time was 0.738474722 seconds
student@student-virtual-machine:~/Desktop/Summer-2024-CSC-415-assignments/csc-415-assignment-4-word-blast-RobelKasahun$
```