Nathan Rennacker ID: 921348958
Github: nlrennacker CSC415 Operating Systems

# Assignment 4 - Word Blast

**Description:**

This program reads the entirety of War and Peace using an optional number of threads to break up the txt file in order to make it run faster. After counting the occurrences of every word, the top 10 are printed to the terminal.

**Approach / What I Did:**

To start off I decided I needed to understand threading and spent a fair amount of time reading through the documentation and rewatching the lectures.
Then I started planning the data structures needed. These included a data structure to store the most frequent words, with the struct holding two fields - a character pointer to store the word and an integer to store the count, various global variables for the thread function to use, and the mutex lock which was going to be used by all the threads.
In terms of function, the program then opens the file given in the argument and determines the file size using lseek. It then calculates the block size based on the number of threads, with the file divided equally among them. Next, the program creates a thread for each block of the file and passes an ID (simple int) to each thread to identify the starting point for that thread. Each thread processes a block of the file and counts the frequency of each word, incrementing the count in the data structure when a word is found. After all threads complete, the program sorts the data structure in ascending order of count and prints the top ten most frequent words with their counts.

**Issues and Resolutions:**

Initially I had trouble getting pread to work correctly until I realized I needed to pass a unique number to each thread (which I did using the args pointer in the passed thread function) that helped calculate the starting point in the file where pread was supposed to start reading.

I tried to implement a hashmap for this program and got most of the way completed but I couldn't figure out why some parts weren't working. So I ended

up deciding that, for the scope of this project, it wasn't worth spending a ton of time on the hashmap and just used a simple array instead. (The sorting was going to be similar in either case)

**Analysis:**

Obviously, as you increase the number of threads, the total time to complete the program decreases–up to a point. The time it takes to execute stops decreasing as you reach more than four threads. It looks like the virtual machine is limited in the number of cores it can use, each core can support up to two threads and that seems to be the maximum number used for the Word Blast program as the time it takes to complete the search decreases from 1 to 2 cores but stops decreasing after that.

**Compilation**

```
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ make
gcc -c -o Rennacker_Nathan_HW4_main.o Rennacker_Nathan_HW4_main.c -g -I.
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ S
```

**Execution**

**One Thread**

```
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ make run
./Rennacker_Nathan_HW4_main WarAndPeace.txt 1


Word Frequency Count on WarAndPeace.txt with 1 threads
Printing top 10 words 6 characters or more.
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1928
Number 3 is Natásha with a count of 1213
Number 4 is Andrew with a count of 1143
Number 5 is himself with a count of 1020
Number 6 is Princess with a count of 916
Number 7 is French with a count of 881
Number 8 is before with a count of 833
Number 9 is Rostóv with a count of 776
Number 10 is thought with a count of 767
Total Time was 2.509319434 seconds
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$
```

Nathan Rennacker
Github: nlrennacker

ID: 921348958
CSC415 Operating Systems

### Two Threads

```
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ make run
./Rennacker_Nathan_HW4_main WarAndPeace.txt 2


Word Frequency Count on WarAndPeace.txt with 2 threads
Printing top 10 words 6 characters or more.
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1928
Number 3 is Natásha with a count of 1213
Number 4 is Andrew with a count of 1143
Number 5 is himself with a count of 1020
Number 6 is Princess with a count of 916
Number 7 is French with a count of 881
Number 8 is before with a count of 833
Number 9 is Rostóv with a count of 776
Number 10 is thought with a count of 767
Total Time was 1.441155139 seconds
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$
```

### Four Threads

```
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ make run
./Rennacker_Nathan_HW4_main WarAndPeace.txt 4


Word Frequency Count on WarAndPeace.txt with 4 threads
Printing top 10 words 6 characters or more.
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1928
Number 3 is Natásha with a count of 1212
Number 4 is Andrew with a count of 1143
Number 5 is himself with a count of 1019
Number 6 is princess with a count of 916
Number 7 is French with a count of 881
Number 8 is before with a count of 833
Number 9 is Rostóv with a count of 776
Number 10 is thought with a count of 767
Total Time was 1.348665460 seconds
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$
```

### Eight Threads

```
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$ make run
./Rennacker_Nathan_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 1963
Number 2 is Prince with a count of 1928
Number 3 is Natásha with a count of 1213
Number 4 is Andrew with a count of 1143
Number 5 is himself with a count of 1020
Number 6 is princess with a count of 916
Number 7 is French with a count of 881
Number 8 is before with a count of 833
Number 9 is Rostóv with a count of 776
Number 10 is thought with a count of 767
Total Time was 1.379086568 seconds
student@student-VirtualBox:~/Desktop/Spring23Assignments/csc415-assignment-4-word-blast-nlrennacker$
```