Russell Azucenas
917422693
GitHub ID: Ninjaruss

Assignment 4: Word Blast

https://github.com/CSC415-Fall2020/assignment-4-word-blast-Ninjaruss

Description:

This program is designed to read a text file and output the top ten most frequent words that appear in the file. The program should implement multithreading with appropriate critical section protection. For the counting of frequency, this program is **case-sensitive** and will differentiate between lowercase and uppercase words.

Process:

I divided the plan into three sections: data structure implementation, file processing, and thread creation. For the data structure I used a linked list originally, but it was changed into a hash map for the sake of efficiency (see Note below). The hashmap structure was created via a Word struct containing token and frequency with a WordList struct to hold a pointer list of Word nodes. I only needed four functions to implement the hashmap structure: hash, initialize, get, and set. File processing was implemented as a separate processFile function in order for threads to work simultaneously. The function simply tokenizes the chunk of text specified by the thread then populating the hashmap with the word and its frequency. I included a mutex lock for two actions operating on the hashmap; these being a newWordLock and a addFreqLock. The newWordLock is designed to lock the action of adding a new word into the hashmap (read operation) while addFreqLock is designed to lock the action of adding a new frequency to the hashmap (write operation). For thread creation, I simply allocated space for each thread along with a fileChunk struct that determines which chunk of the file to process. I then used pthread_create() to create each thread and waited for each to finish with pthread_join().

Note:

This assignment was mainly tough due to the decision of which data structure to use. I initially opted to use a linked list, but down the line I struggled with trying to sort the list so that I can return the top ten most frequent words. Due to my inexperience with linked lists, I spent way too much time trying to resolve this issue. As a result, I made the decision to switch my data structure to a hashmap since it was essentially a dictionary to me. Despite it costing me a ton of time, I was able to complete the assignment with the newly implemented structure. I left the previous linked list code in the repo if you're interested.

Analysis:

Upon inspection, my program seems to have a completion time within the range of 0.5 and 0.6 seconds between all tested threads. I believe that the time is mainly based on the loop operations when trying to add a new frequency to a word. This is due to the fact that trying to add a new frequency requires the program to search through the entire hashmap to find the specified word. In hindsight, I assume that referencing the hash code position may speed up the completion time. Outside of that, limiting the max size for the hashmap will probably reduce the completion time as well. However, I found that reducing the hashmap's size resulted in certain words not appearing in the hashmap due to collisions with hashes matching each other.

Oddly enough, I found that increasing threads seems to increase the total time by milliseconds on average. I'm not entirely sure of the cause for these differences, but I suspect that these are likely due to either the critical section protection or something to do with how I divided the data among the threads. Despite that, there will be times where more threads will perform better than a single one. The high variance could be attributed to a number of factors such as available cores or possibly my thread join implementation.

Output:

```
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$ make run RUNOPTIONS="WarAndPeace.txt 1"
./Azucenas_Russell_HW4_main WarAndPeace.txt 1
Printing top 10 words 6 characters or more. (case sensitive)
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1577
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 1017
Number 6 is French with a count of 881
Number 7 is before with a count of 779
Number 8 is Rostóv with a count of 776
Number 9 is thought with a count of 766
Number 10 is CHAPTER with a count of 730

Completed using 1 threads.
Total Time was 0.535593313 seconds
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$ make run RUNOPTIONS="WarAndPeace.txt 2"
./Azucenas_Russell_HW4_main WarAndPeace.txt 2
Printing top 10 words 6 characters or more. (case sensitive)
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1577
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 1017
Number 6 is French with a count of 881
Number 7 is before with a count of 779
Number 8 is Rostóv with a count of 776
Number 9 is thought with a count of 766
Number 10 is CHAPTER with a count of 730

Completed using 2 threads.
Total Time was 0.531713593 seconds
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$
```

```
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$ make run RUNOPTIONS="WarAndPeace.txt 4"
./Azucenas_Russell_HW4_main WarAndPeace.txt 4
Printing top 10 words 6 characters or more. (case sensitive)
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1577
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 1017
Number 6 is French with a count of 881
Number 7 is before with a count of 779
Number 8 is Rostóv with a count of 776
Number 9 is thought with a count of 766
Number 10 is CHAPTER with a count of 730

Completed using 4 threads.
Total Time was 0.559217607 seconds
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$ make run RUNOPTIONS="WarAndPeace.txt 8"
./Azucenas_Russell_HW4_main WarAndPeace.txt 8
Printing top 10 words 6 characters or more. (case sensitive)
Number 1 is Pierre with a count of 1963
Number 2 is Prince with a count of 1577
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 1017
Number 6 is French with a count of 881
Number 7 is before with a count of 779
Number 8 is Rostóv with a count of 776
Number 9 is thought with a count of 766
Number 10 is CHAPTER with a count of 730

Completed using 8 threads.
Total Time was 0.558516438 seconds
student@student-VirtualBox:~/Documents/assignment-4-word-blast-Ninjaruss$
```