Output with 1 Thread:

Text

Description automatically generated

Output with 2 Threads:

Text

Description automatically generated

Output: with 4 Threads:

Text

Description automatically generated

Output with 8 threads:

Text

Description automatically generated

Code:

Text

Description automatically generated

My Approach for map-reduce:

I didn’t use any regrex functions since I thought a simple if statement will take care of checking matches.

I definitely didn’t got the expected result, and I owe that to the fact that my locks may be off.

But when I take/erase locks from line 30 and 32 my results are way off. Therefore, I decided to keep them to be “safer”.

For the output, more threads were definitely much faster, but results were way more off.

Some problems I encountered, was the fact that I couldn’t figure out how to sum all instances together of the 8 files. I had the instances of each individual file, but I couldn’t figured out. Until I decided to load every file while counting the instances, that way I only returned the final modified dictionary and not every dictionary.

In my program, I measured how much it takes for the 8 files to be processed/count words and the total time of the whole program. Therefore, there are 8 performance measurements (8 files), the final dictionary and total time of program.

I guess the bugs that might be in my code, is the fact that total instances doesn’t match with the stated ones in teams.

Assignment took me about 5/6 hours

**CPU info:**

model name : Intel(R) Core(TM) i7-1065G7 CPU @ 1.30GHz

4 36 220