Alejandro Paz
Parallel Computing

Map Reduce MPI version

Instructions to Run
To run you have to do mpirun -n <# of threads> python3 map_reduce_MPI.py

What problems did you encounter completing the assignment and how did you overcome them?

I'm not experienced with MPI at all and had to do a lot of research in order to overcome some problems such as the recv() not working properly. I also had to learn how to use the rank and size to make the program work properly. I had some minor issues as well such as my total time not working but that was because my time.time was not time.time()

Any problems you weren't able to overcome or any bugs still left in the program?
Yes, 2 threads seem to either take too long or not run at all, 8 threads don't run at all and 4 threads give me multiple times.

About how long it took you to complete the assignment

I would say 2 days. 1 day I worked on it a bit but didn't put all my effort into it. The other day I put all my time into it.

Performance measurements (given in seconds) for 1, 2, 4, and 8 threads

```
student@linux-4oae:~/Documents/Python/map-reduce-apaz7> mpirun -n 1 python3 map_reduce_MPI.py

{'hate': 332, 'love': 3070, 'death': 1016, 'night': 1402, 'sleep': 470, 'time': 1806, 'henry':
 661, 'hamlet': 475, 'you': 23306, 'my': 14203, 'blood': 1009, 'poison': 139, 'macbeth': 288,
'king': 4545, 'heart': 1458, 'honest': 434}
Total time:  2.7405054569244385
```

```
student@linux-4oae:~/Documents/Python/map-reduce-apaz7> mpirun -n 4 python3 map_reduce_MPI.py

None
Total time:  0.2476353645324707
None
Total time:  0.2932438850402832
None
Total time:  0.6612865924835205
{'hate': 332, 'love': 3070, 'death': 1016, 'night': 1402, 'sleep': 470, 'time': 1806, 'henry':
 661, 'hamlet': 475, 'you': 23306, 'my': 14203, 'blood': 1009, 'poison': 139, 'macbeth': 288,
'king': 4545, 'heart': 1458, 'honest': 434}
Total time:  0.6471495628356934
```

A short analysis of why the program behaves as it does with an increasing number of threads

The more threads you have all working to accomplish a task then faster your program will complete. With only one thread in play, finishing the count of words across multiple files will take longer than when you have multiple threads that could all count the number of words at the same time.

Any observations or comments you had while doing the assignment

It seems to me a little harder to do it this way but that might just be my inexperienced with MPI

Output from the cpuInfoDump.sh program

```
model name» : AMD Ryzen 7 4700U with Radeon Graphics
      4      40     208
```