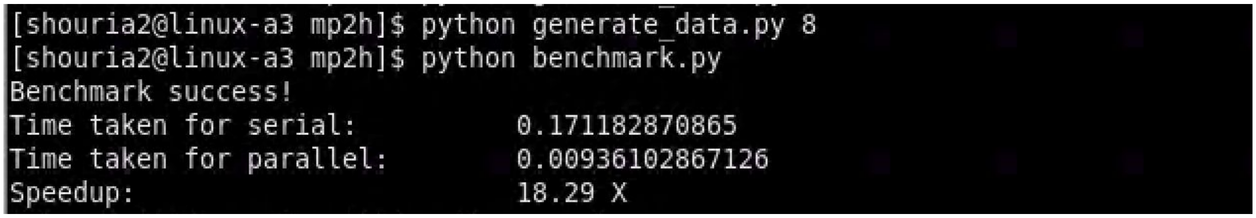
Shouri Addepally
ECE 220 Honors
03/31/2020

Questions:

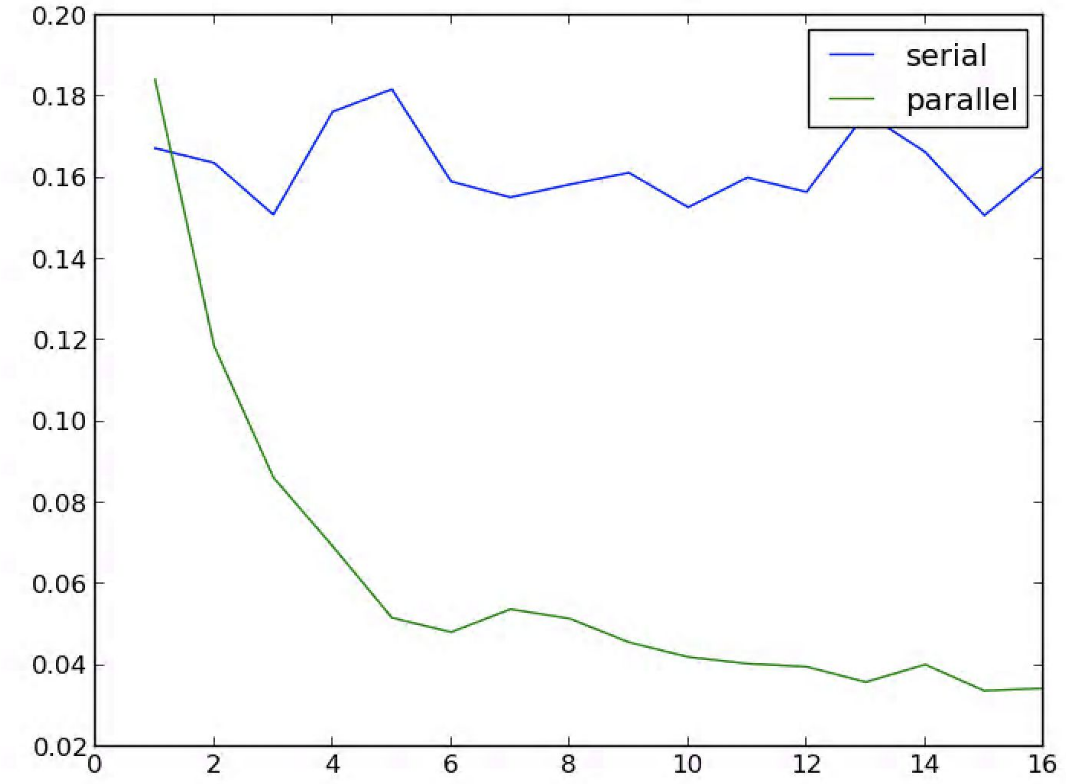
6. If we don't use lock_for_result in acc.h to guard the result, then when we run acc_parallel we will not get consistent results for a variety of times that are tested. The reason for this is because without the designated mutex lock, there will be multiple threads that will each try to access the value at a particular memory location at a given result. With these multiple threads trying to access the memory location, that value will either get overwritten or be modified in a way which will turn out to be incorrect. If we use a mutex lock to protect the result, there will only be a singular thread which has access to it and this will allow the correct sum to remain.

7.

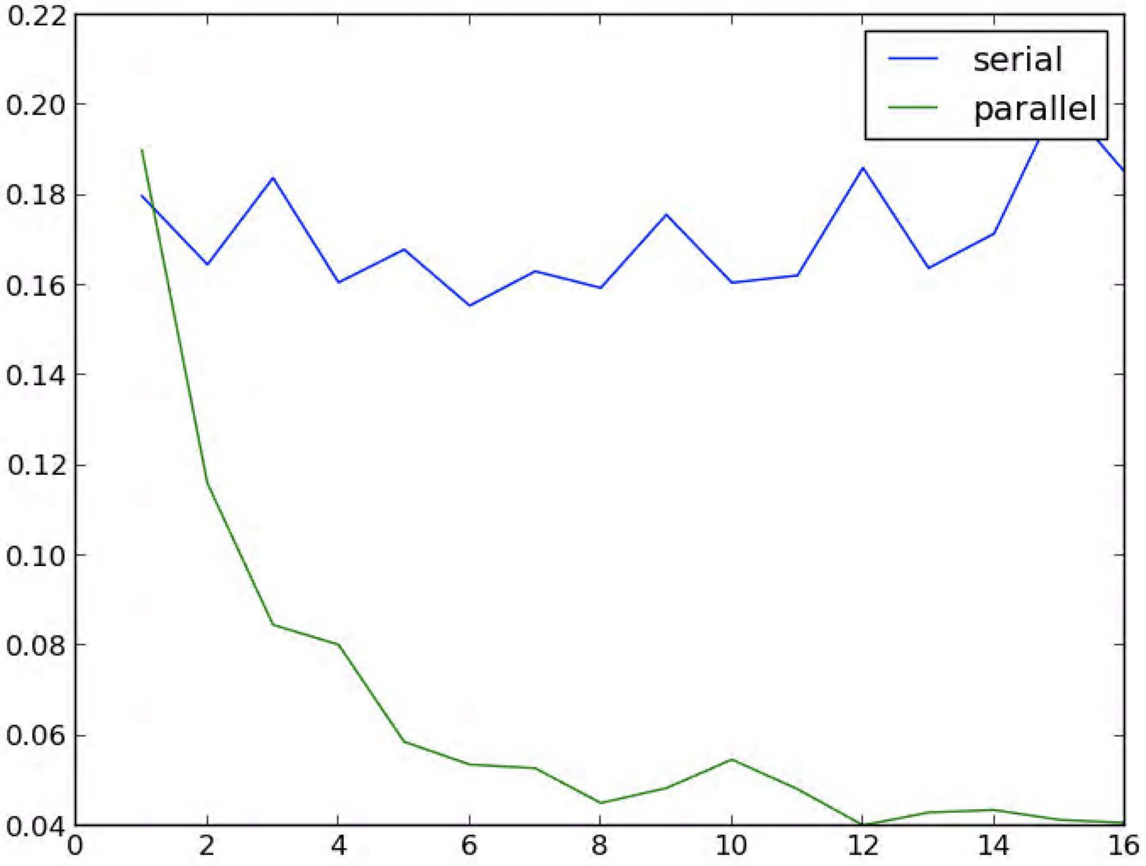


```
[shouria2@linux-a3 mp2h]$ python generate_data.py 8
[shouria2@linux-a3 mp2h]$ python benchmark.py
Benchmark success!
Time taken for serial:          0.171182870865
Time taken for parallel:        0.00936102867126
Speedup:                        18.29 X
```

8. MPH1:



MPH2:



From both of the graphs, we can see that they are relatively similar. However, the approach that was followed in MP2H is better because we use synchronization which can incrementally keep track of the overall sum and allow the boss thread to automatically output the final answer at the end of the program. However in MP1H, we needed the boss thread to use partial sums to calculate the final result. MP2H was more efficient and was a better option because it didn't require the boss thread to keep performing computations and allowed for the final result to be outputted using the boss thread instead.