Group 16

Group Members: Emmy Woods, Chris Cornell, Brandon Roth

From The README

Matrix Multiplication:

how to build and run matrix multiplication:

the sequential version:

gcc mmseq.c -o mmseq

./mmseq

the OpenMP version:

gcc mmomp.c -o mmomp -fopenmp

./mmomp

the pthread version:

gcc mmpthread.c -lpthread -o mmpthread

./mmpthread

How to test:

time ./mmseq

time ./mmomp

time ./mmpthread

The is a constant defined for the size of the matrix called SIZE that needs to be

changed before testing. In the pthread version, there's also a constant for the

number of threads which can be changed.

There is also code for printing matrices that is commented out.

Uncomment that to print matrices to check correctness.

* hashing algorithm
  + Sequential
    - The sequential algorithm has 4 hardcoded loops that pass permutations of each letter in loop one into the passing function to fund a match to the passed in hash. This is done in a sequential matter. Permutations of ‘a’ being the first letter will be relatively quick but a hash with z as the first letter may take a while to find since it is the last element in the list
  + parallel
    - The parallel version creates a thread for every permutation up to the limit of threads. So with 10 threads, anything with in the first 10 threads will happen very fast but anything after that will be evaluated a lot slower.