Jess Conway
February 12, 2018
COMP 460; Sekharan
Homework 2

I have performed both of these tasks using static scheduling of threads. To my understanding, static scheduling is best used in situations where all threads essentially perform the same task, while dynamic scheduling is best suited for situations where the workload may vary depending on which thread one is analyzing. For that reason, static scheduling seemed best for these tasks because I used a very basic algorithm for each, and simply partitioned these algorithms into smaller parts when using multiple threads.

I have attached code and graphs for each problem and associated graphs which compare various runs of the programs using different sized data sets and different numbers of threads. These are labelled as appropriate.

One will notice from the matrix graphs that the speedup from multithreading is most obvious when multiplying very large matrices. This is appropriate, as larger matrices, even if sped up by the same factor as smaller matrices, would allow for a more notable effect on time-decrease.

These same trends also appear when we multithread the code for factoring large numbers. As previously mentioned, because I stuck with the basic algorithm and this code is designed for large numbers, I used static scheduling to parallelize the code. While this actually led to a time increase for smaller numbers (because creating threads takes time), this increase was negligible and is acceptable given the time decrease when factoring larger numbers. Once again, the larger the number was, the more profound the reduction in time became.

As an exercise, I also attempted to create a program that used dynamic scheduling on the factoring problem. While this code also resulted in significant reduction in the time needed to factor a number, the results were not always accurate and therefore led me to believe I have some sort of misunderstanding about how dynamic scheduling works. I've attached this code as well in case it is of interest.

Digital copies of all of my work can be found here if needed:
http://github.com/ConwayJ18/comp460s18hw2