**CSC 338 Parallel and Distributed Programming**

**Test 2**

Copy the Test2 folder from trace to your computer. Add your name in a comment at the top of test2.c and complete test2.c to read a file of numbers and calculate and print the sum of the numbers using n threads or processes, where n is a command line argument. For example, the program will be invoked as follows for four threads:

./test2 4

or, for MPI with four processes:

mpiexec ./test2 –n 4

Also print the time taken to perform the calculation. This should not be the time taken by the entire program (for example, don't time how long it takes to read the file) but it should isolate the time taken to create the threads and sum the numbers. The time will be very small so use scientific notation (“%e”) when you print it.

You are given code that includes a function to read the file of numbers and another function that returns how many numbers are in the file. The program's main function has some variable declarations; those are not guaranteed to be all the variables you need but they are the ones I used. You are also given three files of numbers – randnums40.txt has 40 random numbers, randnums500.txt has 500 numbers, and randnums.txt many more numbers. count\_nums.py is a serial Python program that can help you check your results.

You may use Pthreads, OpenMP, or MPI. If you use OpenMP, you should time your code using omp\_get\_wtime() . If you use Pthreads, you should use timer.h. If you use MPI, use MPI\_Wtime().

The test will be graded according to the following criteria:

Name in comment at top of program………………………………………………. 10%

Basic structure with main function……….………………………………………... 20%

Read command line argument………………..…….……………………………… 20%

Calculate total with one thread……….…………………………..………………... 10%  
Correctly create n threads…...………….………………………………………….. 10%

Correct implementation for 4 processes and 40 integers…..………………............. 15%

Correct implementation for arbitrary numbers of threads and integers.……............ 15%

Copy the Test2 folder, with your completed program, to your upload folder on trace; be sure your name is in a comment at the top of the program.

This is an open-book/open-notes test; you may use the text, your notes, and anything on trace.