

## Assignment 2 – OpenMP

Physics 7411 – Computational Physics: Computing for Petascale Systems

Due: March 25, 2009 at 6:00 p.m Central time

The purpose of this assignment is to get an introduction to shared-memory programming with OpenMP. This assignment is based on the problems given in the OSC tutorial *Parallel Programming with OpenMP*, which you can view at <http://oscinfo.osc.edu/training/openmp>

1. Write a program where each thread prints the message “Hello World!” along with its thread ID number and the total number of threads used. Run with 4, 8, and 16 threads.
2. Write a program that declares an array A to have 16000 integer elements and initialize A so that each element has its index as its value. Then create a real array B which will contain the running average of array A. That is:
  - i.  $B(i) = (A(i-1) + A(i) + A(i+1))/3.0$except at the end points of B. Your code should do the initialization of A and the running average in parallel using 8 threads. Experiment with all four of the scheduling types for the running average loop by timing the loop with different schedules. Use static scheduling for the initialization loop.
3. Write a program to read in the x,y,z coordinates from the file `points.dat` (which will be posted on the course website soon) and calculate the geometric center which is the average x value, the average y value and the average z value. Do the calculation in parallel for x, y and z using the parallel sections construct. The `points.dat` file is ASCII with one x,y,z triplet per line. Calculate the average coordinate value given by the equation  $(\sum x_i + \sum y_i + \sum z_i)/3N$ , where N is the number of points (lines in the input file). Implement using a shared global sum and a critical region.
4. Modify the program in 3 to compute the geometric center using a single for loop and loop-based parallelism. Compute the global sum for average coordinate value using the reduction construct.

You may work individually or in groups of 2 on this assignment. You should turn in a copy of the OpenMP programs for 1-4 above.