CS 3123 (McNally) : High Speed Computing Winter 2023

Assignment # 2 : DUE Monday February 6^{nd} by 3:30pm in the assignment slot (or in lab with e-copys of your files also submitted as a back up to D2L). Please staple these pages as the cover page to your physical assignment with your full UNB name PRINTED to the top left corner. Please staple your assignment in the top RIGHT CORNER. Follow the guideline document posted in D2L for programs. You should be able to run these on cedar.computercanada.ca or the lab

For each of the following questions:

a. Execute this program (with timers) for N = 500 with 4 processes; Run the program 14 times and calculate the trimmed average time (removing the lowest 2 and highest 2 times obtained). Determine the standard deviation of the remaining times.

b. Execute this program (with timers) for N = 1000 with 4 processes; Run the program 14 times and calculate the trimmed average time (removing the lowest 2 and highest 2 times obtained). Determine the standard deviation of the remaining times.

c. Execute this program (with timers) for N = 2000 with 4 processes; Run the program 14 times and calculate the trimmed average time (removing the lowest 2 and highest 2 times obtained). Determine the standard deviation of the remaining times.

d. Execute this program (with timers) for N = 10,000 with 4 processes; Run the program 14 times and calculate the trimmed average time (removing the lowest 2 and highest 2 times obtained). Determine the standard deviation of the remaining times.

e. Create a table of the resulting averages and standard deviations. Use regression (excel does this easily - just ask me) to predict the relationship between time and N and plot both your data and the line on a single plot (again excel does this).

1. Write a parallel program using OPENMP (C or Fortran) which will (for different n values) make an array of random integers from 1 to 10 and create a frequency table of how many (both count and relative count) of each value appeared. Have your program display relative count as a decimal with 4 decimal places.

2. Write a parallel program using OPENMP (C or Fortran) which will (for different n values) determine

$$\sum_{i=1}^{n} \frac{x_i}{x_i + 1} * (-1)^i$$

3. Write a parallel program using OPENMP (C or Fortran) which will (for different n values), create two random double valued matrices of size N and add them. Observe the limits of N and just report those cases as un-executable appropriate.

NOTE: in future weeks we also would need a sequential program for the same problems and look at the sequential timings compared to the parallel timings (we only one one parallel time for each program execution - not one from each processor)