Assignment MPI: Numerical Integration : Collective

The purpose of this assignment is for you to learn more about

- implementing a static workload partitioning scheme in MPI on numerical integration using a reduction collective,

As usual all time measurements are to be performed on the cluster.

To be able to compile and run an MPI program on Centaurus, **you need to add the line** `module load openmpi/4.1.0` **at the end of the file** `.bashrc` **located in the home directory of your account on Centaurus**. (log off and back in afterward.)

To compile an MPI application, use the `mpicc` compiler in C and the `mpicxx` compiler in C++. They also serve as linker. To run an MPI application using 19 processes, you can run `mpirun -n 19 ./myprogram`. But you will need to have a proper node allocation first. And if you have a proper node allocation then specifying `-n` is not necessary because the cluster scheduler does that for you.

1 Numerical Integration : Collective (35 pts)

Adapt the numerical integration application to make it use MPI in a simple way. The first MPI process should take the first N/P iterations of the loop, the second should take the next N/P iterations of the loop, etc.. The partial integration should be accumulated on rank 0 so that it can print the correct answer to `stdout` and the time it took to `stderr`.

Question: Go into the `num_int/` directory. Write the code in `mpi_num_int.cpp`. You can test your code with `make test`.

Question: Run and time that program on Centaurus using many configurations. Use the `make bench` to queue all the jobs.

Question: Generate figures of the results using `make plot`. Do you observe speedup higher than can be achieved on a single machine?