**Program 4 Report**

1. Problem Statement

Using a working copy of the 1D Filtering program without MPI capability, prior to using Bridges, make certain that your code is working correctly on cs-comet . Afterwards, run the program using a single core on Bridges. Then add OpenACC directives to execute the program with one K80 and separately with one P100. Run each of the four configurations (cs-comet one core, Bridges one core, Bridges K80, and Bridges P100) ten times. Use the minimum times for each configuration to determine the speedup with respect to the Bridge one core solution.

1. Approach

I created a .cpp file without mpi capability using pragma openacc to capture the time. Then I created 3 slurm scripts that run on bridges for one core, k80 and p100.

1. Solution
2. Create a slurm file for cscomet, p100, k80 and bridges
3. Integrate the slurm script to execute
4. Add openacc and remove mpi from copied code from program 3
5. Added wtime() to track triple loop and entire program for elapsed time.
6. Run program on cscomet and bridges xsede to get results.

Error: my password was not working for the portal to xsede/bridges