BLG562E – Parallel Computing for GPUs using CUDA
Homework #2
Due on: 27/03/2023

For this homework, you are given a code template to be extended further to implement vector addition in steps 1 and 2 explained below.

1. *Performance of Vector Addition on CPU (sequential code)*

As the first step, you are expected to implement the vector addition code using C as a sequential code. You can compile your code enabling compiler optimizations. Next, run your code and report the execution time. Remember you should do several runs (I expect at least five) and report on the average execution time.

2. *Performance of Vector Addition on GPU*

a) In this part of step 2, implement the vector addition kernel and the host code to run the GPU kernel with only a single block. Compile your code with ***nvcc*** compiler and run to report the execution time. Again you need to make several runs and report on the average execution time. Your code should also include a function to verify the results generated by GPU. You can compare the results generated on the CPU and the results generated on the GPU.

b) In this part of step 2, implement the vector addition kernel and the host code to run the GPU kernel with blocks having a single thread. Compile your code and run to report the execution time.

c) In this part of step 2, change the vector size to 2096; implement the vector addition kernel and the host code to run the GPU kernel with blocks having 128 threads. Compile your code and run to report the execution time.

   Does your code generate correct results?

d) Now, modify your code to run with a specified block size without generating incorrect results. Compile your code and run to report the execution time. Make sure your GPU kernel generates correct results.

3. *Performance Study of Matrix—Matrix Addition*

In this step, implement the matrix-matrix addition as a function to run on CPU, as a GPU kernel to work with a specified block size, and a verification function to compare the results generated by CPU and the GPU. Measure the execution times of the CPU and the GPU code and report on the Speedup. For part 3 and 4, create, use and submit the same source file(s).

4. Scalability Study (Optional, Extra Points)

Study the scalability of matrix-matrix addition by changing the input matrix size (defined by N in the source code) and reporting on the speedups.

In your homework report, start with explaining your methodology in general; describing the compiler and its version that you use, the CPU and GPU processor specifications and the details of your experiments (how many runs you performed, how you calculated the speedup, etc.). For each step briefly describe what you have done and report on the performance.