ECE5640 **High Performance Computing** **Spring 2016**

Homework 6: Due Wednesday, April 6

This homework asks you to implement and examine tradeoffs between different implementations of Sobel Edge detection. The source C++ code and input files are in a shared area on the discovery cluster, as described in Lecture 16 (March 23).

Please submit *two* files on Blackboard:

1. A zip file containing all your code.
2. A pdf file of your report that describes your optimizations and results.

1. (15 points) Use OpenACC to write your Sobel code.
2. (25 pts) Use CUDA to write your Sobel code. You may invoke your CUDA code from C host code or from MATLAB PCT. This choice is up to you. Your CUDA code should use global memory.
3. (45 pts) Rewrite your CUDA code to optimize your Sobel code. You should use shared memory, and any other techniques to make your code run as fast as possible. In your report, describe the techniques you used, and the speed up you obtained over the base implementation. Explain what optimizations worked and why.
4. (15 pts) Time your different implementations. Which one runs fastest? Which was easiest to code? Explain advantages and disadvantages of each. Explain what you would do to run your code more efficiently.
5. (Bonus) Try different sizes of blocks and numbers of thread per block. Describe your choices, reasons for your choices, and how effective they were. What happens if you use more threads versus more blocks? What if you do more work per thread? Analyze your results.