CSc 746 Project 6
OpenCL Jacoby relaxation

Due Tuesday 5/13/2014 by Noon
(75/100 for correctness, 25/100 for documentation)
(*Points may be deducted for horribly inefficient code or poor documentation!*)
(5% of your grade, No late projects accepted)

This assignment is to be done individually!

You have already been given a serial Jacoby/finite difference program in

>   ~whsu/csc746/Code/Fd

which you parallelized using CUDA in the previous assignment. For this assignment, you are to take your team's parallelized code and translate it into OpenCL. Even though you're using the code from your team, you are to do this assignment individually and submit it individually.

Parallelize Implementation 1 (50/75):
Translate the global memory version (Implementation 1) from the previous assignment, into OpenCL.

Time Implementation 1 (25/75):
Time the OpenCL and CUDA versions per the instructions under Measurements, below.

For more information on how to use OpenCL event timers see
>   http://parallelis.com/how-to-measure-opencl-kernel-execution-time/

Extra Credit (Up to 5% of your overall grade):
Translate the shared memory version (Implementation 2) from the previous assignment, into OpenCL. Time the shared memory version per the instructions below.

Measurements
For the OpenCL implementations, get timestamps from just before the first memory copy from host to compute device, and right after the last memory copy from compute device to host. For this project, following the procedure outlined in previous projects, report the execution times for input data set G3202, for two system configurations, for each implementation:

Tesla C2050 (-q fermi.q): W = 32
GeForce 9800 (-q g92.q): W = 16

Hence, there should be two sets of execution times for the assignment, and two sets of execution times for the extra credit.

Report (25/25):
Your report should be a *minimum* of two double-spaced pages, *excluding* tables of data and graphs. Be sure to have an introduction, clearly explain what you did, then present the results and then conclusion.   Be sure your data is integrated into your report document, and clearly labeled. Be sure your graphs are integrated into your report document, with a clear title and accurate labels on all axes.

Submission:
Submit your code and report in a single zip file via the iLearn submission link.