Hsiang Lo

**Project #6**

**Goal**: To learn about CUDA and its ability for parallelism.

**Requirements:**

* Ranges: xc 0.0 – 2.0, yc 0.0 – 2.0, r 0.5 – 2.0
* BLOCKSIZE of 16, 32 and 64
* NUMTRIALS of 16k, 32k, 64k, 256k, 512k
* Use appropriate units: MegaTrials/Second
* Use CUDA timing

**Notes**:

* This is like project #1 except the numbers have changed, and using CUDA.
* Send Xcs, Ycs and Rs arrays ahead of time. Send them to the GPU where they can be used as look-up tables.

**Commentary Questions**

1. Tell what machine you ran this on

This assignment was ran on MacBook Pro 2017 and the execution was done over Rabbit.engr.oregonstate.edu.

1. Show the tables and graphs
   * Performance vs. NUMTRIALS with multiple curves of BLOCKSIZE

Note: The title should be corrected to Performance: MEGATRIALS/s OVER NUMTRIALS

The multiple curves are 16, 32 and 64 which represents 16k, 32k and 64k of BLOCKSIZE

* + Performance vs. BLOCKSIZE with multiple curves of NUMTRIALS

The multiple curves of 16k, 32k, 64k, 128k, 256k and 512k represents the number of trials plotted along BLOCKSIZE and MegaTrials/s

1. What patterns are you seeing in the performance curves?

Increasing number of trials and larger blocks of data is correlated with the increased in performance along the curves/graph.

1. Why do you think the patterns look this way?

Increase in the number of trials show what CUDA can do with parallelism in data processing while increase in block size shows similar idea in that parallelism is capable of doing single data processing idea and apply it to a large stretch of data(blocks)

1. Why is a BLOCKSIZE of 16 so much worse than the other two?

In the lectures, the professor mention that 32 is a good number to work with in the lecture regarding “warp”. That being said, perhaps 16 is just too small of a blocksize to utilize CUDA efficiently.

1. What does that mean for the proper use of GPU parallel computing?

In order to properly utilize GPU parallel computing, it’s important to understand that both block size and trials matter. The idea is that in order to make GPU parallel computing work for you, it goes back to the idea that GPU parallelism is mainly for data processing. A large and adequate amount of work needs to be done to fully utilize the benefit of it!