CS/EE 217
Extra Credit Lab
CUDA Stream
Due Thursday. May 31 at 11:59:59pm

1) This lab extends Lab 1 (Simple Vector Add) by implementing Vector Add with CUDA Streams to overlap data transfer with kernel computation.

2) Lab 1 Vector Add only operated on 1000 elements. In order to see the benefits of CUDA Stream, we will increase the problem size. First, modify your Vector Add to operate on 1 million elements, using a thread block size of 256.

3) Now, partition your problem across two CUDA Streams, so that the first stream will perform a Vector Add on the first 0.5 million elements, and the second stream perform Vector Add on the second 0.5 million elements.

4) You can verify the behavior of the kernels and streams using Nvidia Visual Profiler (nvvp). Is the behavior what you expect to see?

5) Answer the following questions:

a. What is the speed up between the non-Stream and Stream version of Vector Add? Where do the improvement comes from?

b. How can data transfers be further optimized?

c. Do ordering of various CUDA API calls on the host side matter when implementing streams? Why or why not?

Grading:

Please upload your zipped directory (after cleaning up executables and any unnecessary files) to iLearn. Your submission will be graded on the following aspects.

Correctness (60%)

Efficiency (20%)
* Efficient/Effective implementation is used.

Report (20%)
Answers to the questions above