

## CSCD 439/539 GPU Computing Lab7

### Diff operation on GPU using Shared Memory

**Due Time:** Apr 30th 2014 9:50am. No Late Submissions are accepted.

**Rules:** Your code must use C and CUDA Language. If your program shows a compilation error, you get a zero for this lab assignment.

**Submission:** Wrap up all your **source files** into a single zip file. Name your zip file as *FirstInitialYourLastNameLab7.zip*. For example, if your legal name is Will Smith, you should name your zip file as wsmithlab2.zip.

**Before you leave the laboratory, please show the TA or the instructor how your program works, they will give you a score for this Lab assignment.**

**For archive purpose, please also submit your single zip file on EWU Canvas by following CSCD439-01 Course → Assignments → Lab7 → Submit Assignment to upload your single zip file.**

#### **Problem Description:**

In science and engineering, **diff** is used to calculate Differences and Approximate Derivatives. The definition of **diff** is as follows.

$Y = \text{diff}(X)$  calculates differences between adjacent elements of  $X$ , where  $X$  and  $Y$  are 1D array in this lab. If  $X$  is an array of length  $m$ , then  $Y = \text{diff}(X)$  returns a vector of length  $m-1$ . The elements of  $Y$  are the differences between adjacent elements of  $X$ .

$$Y = [X(1)-X(0), X(2)-X(1), \dots, X(m-1)-X(m-2)]$$

I have provided most of the code in the zip file. You are required to implement the following features and answer the questions.

- 1, Read the provided code and understand the input, the output, and how data is transferred between host and device.
- 2, According to the definition of diff above, please write the kernel to compute diff on GPU device, your kernel must use **shared memory** based on our last lecture.
- 3, The input array contains 18 float numbers. I set the block size to 4, defined on the top of source code as a Macro. Please set up the gridSize, blockSize (**formally called execution configuration**) and launch the kernel you implemented. You are supposed to get the results like the following after your run your program.

The original array is: 4, 5, 6, 7, 19, 10, 0, 4, 2, 3, 1, 7, 9, 11, 45, 23, 99, 29,

The diff array is: 1, 1, 1, 12, -9, -10, 4, -2, 1, -2, 6, 2, 2, 34, -22, 76, -70,

4, How the number of global memory access is reduced when compared with your implementation of lab2?

5, Assuming we are in this scenario where each thread processes one different element in an array, each thread only read one element in the array once. Do you think whether in this case a shared memory is still useful with respect to performance improvement?

6, Related with question 5, can you summarize when (in what scenarios) a shared memory is useful with regard to reducing global memory access?