



Programming Assignment 1 Experiments with Vector Addition

Assigned Date

1/28/2013

Due Date

2/4/2013

Overview

1. Starting with vector addition

a. Code

```
__global__ void add (int *a,int *b, int *c) {  
    int tid = blockIdx.x * blockDim.x + threadIdx.x;  
    if(tid < N) {  
        c[tid] = a[tid]+b[tid];  
    }  
}
```

b. And the call

```
add<<<B,T>>>(dev_a,dev_b,dev_c);
```

2. For this assignment, you will make changes to the vector addition code.

a. Different sizes for the vectors – Replace the static declaration for $a[N]$, $b[N]$, and $c[N]$ with dynamically allocated memory and add keyboard input statements to be to specify N .

- i. Use LARGE vectors (push the memory of the cards, should go to at least 1 million)
- ii. Have versions with and without striding

- b. Different CUDA grid/block structures – Add keyboard statements to input different values for:
 - i. Numbers of threads in a block (T)
 - ii. Number of blocks in a grid (B)
- c. Include checks for invalid input.
- d. Timing -- Add statements to time the execution of the code using CUDA events, both for the host-only (CPU) computation and with the device (GPU) computation, and display results.
- e. Compute and graph the appropriate metrics (runtime, speed-up factor,...).

Project Requirements

- 2 versions of the code:
 - A compiled and running sequential C program
 - A compiled and running CUDA program
- Multiple timings of runs of various sizes.
- Appropriate graphs

Deliverables

- Bring code and output to class for discussion next Monday.
- Have a pdf of your writeup and a zip of your source code emailed to Fred Harris and Lee Barford (DO NOT send binaries).
 - Firstname dot Lastname at ... (Fred is cse, Lee is gmail)