CS6023: GPU Programming

Assignment 1 (7 marks)

Submission deadline: Feb 21, 2021, 23:55 on Moodle

Problem specification

Write three separate CUDA C++ kernels for adding up two integer matrices i.e. add elements at the same *i, j* position in both the matrices. In the first kernel *per_row_kernel*, each thread should process a complete row of the input matrices. In the second kernel *per_column_kernel*, each thread should process a complete column of the input matrices. In the third kernel *per_element_kernel*, each thread should process exactly one element from both the input matrices. For the evaluation purpose, *per_row_kernel* will be invoked with *1D grid and 1D blocks*, *per_column_kernel* will be invoked with *1D grid and 2D blocks* and *per_element_kernel* will be invoked with *2D grid and 2D blocks*.

Input: Size of matrix (m and n) and two integer matrices **A** and **B** of same size.

Output: A matrix C of the same size (say $m \times n$) as A and B, storing the result of A+B. C will be provided in the kernel function as a parameter, you need to modify that only.

Points to be noted:

- The file **kernels.h** provided by us contains the prototypes of the three kernels.
- Do NOT change the names and the signatures of the kernels provided.
- Sample input and sample output matrices are shown below. Pay attention to the position of each element in the input and the output matrices.
- The size m and n, of the input matrices used for evaluation will be in the range: $5 \le m \le 2^{13}$ and $5 \le n \le 2^{13}$.
- The updates should be performed on the C matrix and should finally store A+B as result. Do not use any intermediate matrices.
- Do not write any print statements inside the kernel.
- You can use your own main.cu to test your code. We will be using main.cu written by us for evaluating your code.
- Test your code on large matrices.

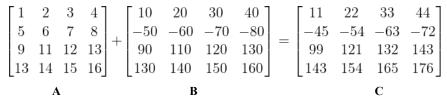


Figure 1: Sample input and output matrices

Submission guidelines

- Submit only one file that contains the implementations of all the three kernels on moodle: https://courses.iitm.ac.in/mod/assign/view.php?id=71500
- The name of the file submitted should strictly be of the format ROLL NUMBER.cu
- For example, if your roll number is CS16D019, the name of the file you submit should be CS16D019.cu
- Make sure that the ROLL_NUMBER part of the filename is in upper case.
- Do not upload anything other than the ROLL NUMBER.cu file.
- After submission, download the file and make sure it was the one you intended to submit.

Learning suggestions:

• Write a CPU-version of code achieving the same functionality. Time the CPU code and GPU code separately for large matrices and compare the performances