

CS6023: GPU Programming

Assignment 1 (7 marks)

Due August 20, 2017 by 23:55 on Moodle

Problem Specification

Implement two separate CUDA kernels for *transposing a 2D matrix*. One kernel, `transpose_parallel_per_row`, should use 1D thread mapping and the second kernel, `transpose_parallel_per_element`, should use 2D thread mapping to accomplish the task. Observe the times taken for single-threaded and parallel executions.

Matrix transpose: For a matrix $A_{m \times n}$, its transpose, $A_{n \times m}^T$ is obtained by interchanging A 's rows and columns.

A sequential implementation of the matrix transpose operation is shown below.

```
for (i = 0; i < m; i++)  
    for (j = 0; j < n; j++)  
         $A^T[j][i] = A[i][j]$ 
```

For this assignment, you are required to download code written by us, and add your own lines of code to it.

You are provided a tarball containing:

1. **main.cu:** Contains the code for allocating memory for matrices on CPU and GPU, kernel invocations and some utility functions.
2. **kernels.cu:** Contains the body (currently empty) of the two kernels. Your task is to implement the two kernels.
3. **kernels.h:** Contains the prototypes of the two kernels.
4. **timer.h:** Contains methods to time your code.
5. **makefile:** Run `make` from the command line to compile your code. It produces an executable with the name `transpose`.
6. **sample.txt:** Contains input and expected output for two sample matrices.

Try launching the kernels with different number of threads and with different matrix dimensions.

Note: Do not change the kernels' signatures and the names of the kernels.

Submission Instructions

When ready to submit,

1. Rename the file `kernels.cu` to `ROLL_NUMBER.cu`.
For example, if your roll number is `CS14D406`, your file should be called `CS14D406.cu`
2. Upload `ROLL_NUMBER.cu` on moodle: <https://courses.iitm.ac.in/course/view.php?id=837>
3. Download your file, and make sure it was the one you intended to submit.