GPU Lab

COSC 6060

Spring 2022

Server hostname is everest.mscsnet.mu.edu

Username and Password is same as pascal

In case your login does not work, use this account as a backup

Username as pdc\_shared

Password as marquetterocks

**Problem 1**: This problem is about understanding basic 1D thread creation.

Use hello.cu program for this problem. Run the program and inspect the output.

What is the output of the program when number of threads (num\_threads) is 21.

**Problem 2**: This problem is about comparing 1D vs 2D thread creation. Use hello2D.cu program for this problem.

1. Kernel function HelloCuda1D(): What is the output of this function when number of threads in a block is 4 and number of blocks in 3. How many threads get created?
2. Kernel function HelloCuda2D(): What is the output of this function with the following thread configurations. How many threads get created?

dim3 block(3, 2);

dim3 grid(1, 2);

**Problem 3**: After understanding how 2D thread organization works, let us give some work to GPU threads to do pixel manipulation of an image. This problem is about color image to grayscale image conversion. Use “colorToGrey.cu” program for this problem. An image is provided to test the code. README file shows how to compile this program.

1. Run the program on Everest server and copy the greyscale image generated by the program to your laptop. Upload that image to D2L lab folder.
2. Change the function “rgb2grayincuda”, so that only half the image becomes grey, the other half simply becomes black.

|  |
| --- |
| Normal greyscale image generated by the program |
| **New greyscale image to be generated by your program** |

Read the first four pages of the attached pdf to get a background about this problem.

**Hint**: For a [grayscale images](https://homepages.inf.ed.ac.uk/rbf/HIPR2/gryimage.htm), the pixel value is a single number that represents the brightness of the pixel. This single number is an 8-bit integer giving a range of possible values from 0 to 255. An unsigned char requires 8 bits of storage in C. Typically zero is taken to be black, and 255 is taken to be white. Values in between make up the different shades of gray.