SCHOOL OF COMPUTER SCIENCE
COURSEWORK ASSESSMENT PROFORMA

MODULE & LECTURER: CM3103, Professor David Walker

DATE SET: 7 November 2018

SUBMISSION DATE: 7 December 2018 at 9:30am

SUBMISSION ARRANGEMENTS: See below

TITLE: Programming with CUDA

This coursework is worth 10% of the total marks available for this module. The penalty for late or non-submission is an award of zero marks. You are reminded of the need to comply with Cardiff University's Student Guide to Academic Integrity. Your work should be submitted using the official Coursework Submission Cover sheet.

INSTRUCTIONS

In this coursework you will modify an existing sequential program in C to produce a parallel version of the code that uses CUDA. You will then run your CUDA code on a machine with an NVidia GPU installed, such as the machines in the Linux lab. You will time different sections of your code and discuss any opportunities for optimising it. In addition to the lecture notes, you may find NVidia's "CUDA Zone" web site: https://developer.nvidia.com/category/zone/cuda-zone.

Here's what you need to do:

1. Download the sequential code, blur.c, and the input file, David.ps, from Learning Central.
2. Log into your account on one of the machines in the Linux lab (or some other machine with a CUDA-capable GPU – see https://developer.nvidia.com/cuda-gpus), and copy blur.c and David.ps to your file space (for example, using the scp command).
3. Edit blur.c to correctly refer to the input and output files in your file system.
4. Compile and run blur.c several times (say, between 6 and 10 times), noting the time for the execution of the main computational work. Evaluate the average and standard deviation of these times.
5. Produce a parallel CUDA version of blur.c named blurCUDA.cu. Edit the code so that the time spent on the following is measured:
    a. Reading in the input file.
    b. Allocation of device memory.
    c. Transferring data between host memory and device memory.
    d. Doing the blurring.
    e. Outputting the blurred image.

    Run the code several times and record the average time, and the standard deviation, for each phase. Vary the value of nblurs (try 10, 20, 40, 80, and 160) and check if the time per blur is a constant.

    [3 marks for correct CUDA code]

6. Write a report (2 or 3 pages of text, plus figures) that presents and interprets the results of your timing experiments. The report should include:
    a. A description of the hardware and software environment.

    [1 mark]

    b. A description of the timing experiments carried out.

    [1 mark]

    c. Appropriate graphs of your timing experiments.

    [2 marks]

    d. A discussion of the results that accounts for any unusual features, and suggests ways to improve the performance of the CUDA program.

    [2 marks]

    e. A short section presenting any overall conclusions, and giving a reflection on what you have learned from this coursework.

    [1 mark]

SUBMISSION INSTRUCTIONS

All submission should be via Learning Central unless agreed in advance with the Director of Teaching.

Students should submit all their files as a single zip (*.zip) file.

Description		Type	Name
Cover sheet	**Compulsory**	One PDF (.pdf) file	[student number].pdf
Report	**Compulsory**	One PDF (.pdf) file, including any relevant plots	report_[student number].pdf
Code	**Compulsory**	One or more source files	No restriction
Input image file	**Compulsory**	One Postscript (.ps) file	David.ps
Output image file	**Compulsory**	One Postscript (.ps) file	DavidBlur.ps
Output data file	**Optional**	One Excel (.xlsx or .xls) file	Spreadsheet of all your timing data

CRITERIA FOR ASSESSMENT

See instructions above for mark scheme.

Feedback on your performance will address each of these criteria.

FURTHER DETAILS

Feedback on your coursework will address the above criteria and will be returned in approximately 2 weeks.

This will be supplemented with oral feedback via individual appointment