

CS426/525 Spring 2020

Project 4

Due May 22, 2020 23:55

1. General Description

In this project, you will explore capabilities of CUDA environment. Project is composed of two parts. First part includes several questions that you will answer. Second part is a programming assignment.

2. Questions

- a. What is control flow divergence?
- b. How can we create a dynamic sized shared memory?
- c. How can we use shared memory to accelerate our code?
- d. Which CUDA operations give us device properties? To answer this question you should write a simple program and query the device properties of the machine you are working with.
- e. What are the necessary compiler options in order to use atomic operations?

3. Programming Assignment

- a. Angle between Two Vectors on GPU: In this part, you will implement a program to find the angle between two vectors using both CPU and GPU implementation. You will compare the execution times of CPU code and GPU code.
 - Your program should take a command line parameter N:
 - N will be the size of the vector.
 - You will write an array generator code which will create two integer arrays with N elements.
 - You will implement a serial CPU code to find the angle between 2 vectors.
 - Second you will implement a parallel GPU code to find the angle between 2 vectors.
 - Try to use information that you have learned from the questions part.
 - Print your results in degrees, not radians, with 3 digits after the decimal point.

4. Output Of Your Program

Your output must be in the following form.

Info

Number of elements: 20000000

Number of threads per block: 32

Number of blocks will be created: 2441

Time

Time for the array generation : 859.039429 ms
Time for the CPU function : 157.895462 ms
Time for the Host to Device transfer : 46.611969 ms
Time for the kernel execution : 26.325537 ms
Time for the Device to Host transfer : 0.013376 ms
Total execution time for GPU : 72.950882

Results

CPU result: 82.126
GPU result: 82.126

5. Submission

- You will submit a single .cu file “angle.cu”.
- You will write a script that can compile and run your code, it should be named as run.sh. Running this script should compile your code with nvcc, run it for the 15 possible cases of your programs, output each as described above.
 - You have to run your code with N=1000000, N=5000000 and N=10000000.
 - You have to run your code with different block sizes. (32, 64, 128, 256, 512).
 - Totally you have 3x5=15 runs.
- Your code also can run with one more argument, a filename. In that file, you will have a number which shows the number of array size and the array elements. For example, if first number is 2000, there will be 4000 numbers, 2000 for each array. This time you will not generate arrays but read from file.
- Therefore your code will run as:
 - If there are 2 arguments (you will use this in your script):
./a N blocksize
 - If there are 3 arguments:
./a N blocksize filename.txt
- You will submit a report that includes:
 - Answers for questions
 - Your implementation details
 - Plot for GPU execution times
 - Discussion of your results
- Email to: berkay.gulcan@bilkent.edu.tr
 - E-mail subject: CS426_HW4
 - You should submit a single .zip file named as name_surname_p4.zip