Lập trình song song
HW0: Làm quen với CUDA

Nên nhớ mục tiêu chính ở đây là học, học một cách chân thật. Bạn có thể thảo luận ý tưởng với bạn khác, nhưng bài làm phải là của bạn, dựa trên sự hiểu thật sự của bạn. Nếu vi phạm thì sẽ bị 0 điểm cho toàn bộ môn học.

Trong môn học, để thống nhất, tất cả các bạn (cho dù máy bạn có GPU) đều phải dùng Google Colab để biên dịch và chạy code (khi chấm Thầy cũng sẽ dùng Colab để chấm). Với mỗi bài tập, bạn thường sẽ phải nộp:

1) **File code** (file .cu)
2) **File báo cáo** là file notebook (file .ipynb) của Colab (nếu bạn nào biết Jupyter Notebook thì bạn thấy Jupter Notebook và Colab khá tương tự nhau, nhưng 2 cái này hiện chưa tương thích 100% với nhau: file .ipynb viết bằng Jupyter Notebook có thể sẽ bị mất một số cell khi mở bằng Colab và ngược lại). File này sẽ chứa các kết quả chạy. Ngoài ra, một số bài tập có phần viết (ví dụ, yêu cầu bạn nhận xét về kết quả chạy), và bạn sẽ viết trong file notebook của Colab luôn. Colab có 2 loại cell: code cell và text cell. Ở code cell, bạn có thể chạy các câu lệnh giống như trên terminal của Linux bằng cách thêm dấu ! ở đầu. Ở text cell, bạn có thể soạn thảo văn bản theo cú pháp của Markdown (rất dễ học, bạn có thể xem ở đây); như vậy, bạn sẽ dùng text cell để làm phần viết trong các bài tập. Bạn có thể xem về cách thêm code/text cell và các thao tác cơ bản ở đây, mục "Cells" (đừng đi qua mục "Working with Python"). Một phím tắt ưa thích của mình khi làm với Colab là ctrl+shift+p để có thể search các câu lệnh của Colab (nếu câu lệnh có phím tắt thì bên cạnh kết quả search sẽ có phím tắt). File notebook trên Colab sẽ được lưu vào Google Drive của bạn; bạn cũng có thể download trực tiếp xuống bằng cách ấn ctrl+shift+p, rồi gõ "download .ipynb".

Đề bài

Câu 1 (1 đ)

Viết hàm và thử nghiệm in ra các thông tin của card màn hình như sau:

8 info

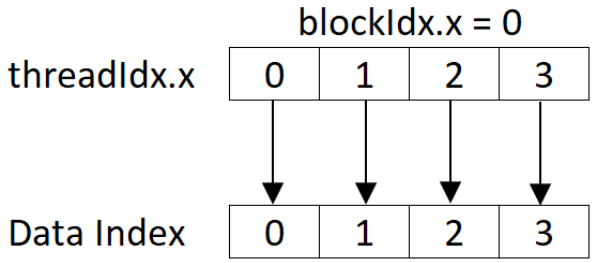
- GPU card's name
- GPU computation capabilities
- Maximum number of block dimensions
- Maximum number of grid dimensions
- Maximum size of GPU memory
- Amount of constant and share memory
- Warp size

Hint: file demo 01-Hello.cu và thông tin của cấu trúc cudaDeviceProp trong đây

Chạy và ghi nhận kết quả trong file "HW0.ipynb", mục "câu 1" đính kèm.

Câu 2 (9 đ)

Trong bài giảng lý thuyết (và file demo 01-AddVector.cu) mỗi thread sẽ thực hiện **một** phép tính cộng trên **một** phần tử của mảng, như hình minh hoạt sau:

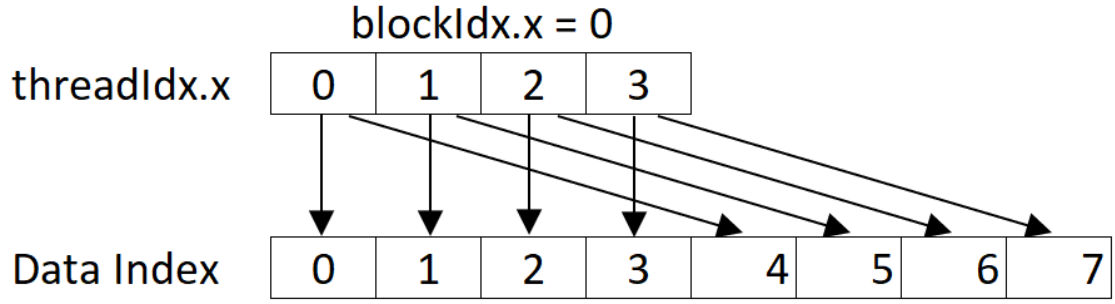


*Hãy viết chương trình cộng hai vector. Tuy nhiên, **mỗi** thread sẽ thực hiện **hai** phép tính cộng trên hai phần tử của mảng thay vì một phần tử như trên.*

Version 1:

Mỗi thread block xử lý 2 * blockDim.x phần tử liên tiếp. Tất cả các thread trong mỗi block sẽ xử lý blockDim.x phần tử đầu mảng, mỗi thread xử lý một phần tử. Sau đó tất cả các thread sẽ chuyển sang blockDim.x phần tử sau của mảng, mỗi thread xử lý một phần tử. Như hình minh họa sau:

Giả sử mỗi block gồm 4 thread.



Gọi in1, in2 là hai mảng đầu vào. Out là mảng đầu ra.

Thread 0: sẽ tính out[0] = in1[0] + in2[0] và out[4] = in1[4] + in2[4]

Thread 1: sẽ tính out[1] = in1[1] + in2[1] và out[5] = in1[5] + in2[5]

Thread 2: sẽ tính out[2] = in1[2] + in2[2] và out[4] = in1[6] + in2[6]

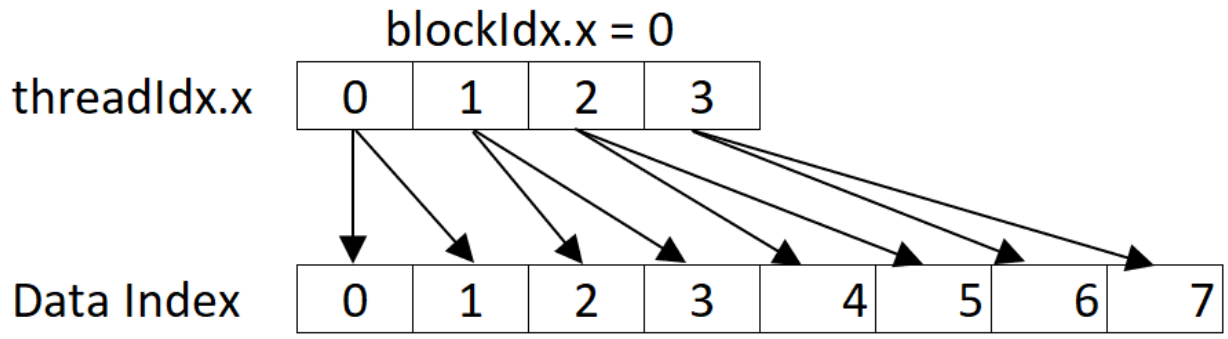
Thread 3: sẽ tính out[3] = in1[3] + in2[3] và out[4] = in1[7] + in2[7]

nhảy blockDim.x phần tử

Version 2:

Mỗi thread block xử lý 2 * blockDim.x phần tử liên tiếp. Mỗi thread sẽ xử lý 2 phần tử liên tiếp nhau trong mảng. Như hình minh họa sau:

Giả sử mỗi block gồm 4 thread.



Gọi in1, in2 là hai mảng đầu vào. Out là mảng đầu ra.

Thread 0: sẽ tính out[0] = in1[0] + in2[0] và out[1] = in1[1] + in2[1]

Thread 1: sẽ tính out[2] = in1[2] + in2[2] và out[3] = in1[3] + in2[3]

Thread 2: sẽ tính out[4] = in1[4] + in2[4] và out[5] = in1[5] + in2[5]

Thread 3: sẽ tính out[6] = in1[6] + in2[6] và out[7] = in1[7] + in2[7]

Code (8 đ)

Sinh viên thực hiện cài đặt cả hai version trên. Để thống nhất SV thực hiện thử nghiệm với kích thước block: 256.

File báo cáo (1 đ)

SV làm trong file "HW1.ipynb", mục "Câu 2": biên dịch file code và chạy với các kích thước mảng **N** khác nhau. Ghi nhận thời gian chạy vào bảng kết quả tổng hợp như sau:

Vector size	**Host time**	**Device time (Version 1)**	**Device time (Version 1)**
64			
256			
1024			
4096			
16384			
65536			
262144			
1048576			
4194304			

16777216			

Nộp bài

SV tổ chức thư mục bài nộp như sau:

- Thư mục <MSSV> (vd, nếu SV có MSSV là 1234567 thì đặt tên thư mục là 1234567)
    - File code “HW0_P1.cu”
    - File code “HW0_P2.cu”
    - File báo cáo “HW0.ipynb”

Sau đó, nén thư mục <MSSV> này lại và nộp ở link trên moodle.