과제 #3

4190.414A 멀티코어 컴퓨팅(001)

Due: 2021년 5월 17일(월) 23시 59분

1 문제 1: OpenCL를 이용한 병렬화 (90%)

행렬 곱셈($A_{M\times K} \times B_{K\times N} = C_{M\times N}$)을 수행하는 예시 코드가 주어진다. 이번 과제부터 GPU를 사용하기 위해 slurm 작업 스케줄러를 사용한다. 로그인 노드에서 `sbatch` 명령을 통해 작업을 실행해야 한다. 다음은 실행 예시이다.

```
[kjp4155@login]$ make
...
[kjp4155@login]$ sbatch run.sh -v -n 3 2048 2048 2048
Submitted batch job 60
[kjp4155@login]$ cat slurm-60.out
Options:
  Problem size: M = 2048, N = 2048, K = 2048
  Number of threads: 1
  Number of iterations: 1
  Print matrix: off
  Validation: on

Initializing matrix... done!
Initializing OpenCL...
Detected OpenCL platform: NVIDIA CUDA
Detected OpenCL device: GeForce RTX 2080
Warming up GPU...2.817663 sec
Warming up GPU...2.780229 sec
Warming up GPU...2.780310 sec
Calculating...(iter=0) 2.780467 sec
Validating...
Result: VALID
Avg. time: 2.780467 sec
Avg. throughput: 6.178771 GFLOPS
```

예시 코드를 수정하여 성능을 높여보자. 주의 사항은 다음과 같다.

- 실습 서버에 탑재된 NVIDIA GeForce GTX 2080 GPU를 사용한다.
- `mat_mul.c`, `kernel.cl` 만 수정할 수 있다. 다른 파일은 채점 시에 예시 코드로 덮어씌워진다. 이전 과제와 달리 `Makefile` 을 수정할 수 없음에 유의하라.
- OpenCL 이외의 병렬화 방식은 사용을 금한다.

- `mat_mul.c`의 각 함수 마지막의 `clFinish` 호출은 올바른 시간 측정을 위해 필요하므로 삭제하면 안된다.
- 성능 측정은 `mat_mul` 함수의 실행시간을 기준으로 이루어지므로, GPU kernel 실행은 `mat_mul` 함수에서만 이루어져야 한다.
- OpenCL 초기화, 메모리 전송 등은 `mat_mul_init` 함수에서 이루어져도 괜찮다.
- 주어진 뼈대 코드를 한번쯤 찬찬히 읽어보고 이해하도록 하자.
- 최적화 힌트: 데이터 레이아웃, 어느 work item이 어느 데이터를 처리할지, work group 크기, local memory 활용, 벡터화 등.
- slurm 작업 스케줄러 사용은 예시에 나온 대로만 해도 되지만, 스케줄러에 대해서도 알아 두면 좋다. `sinfo`, `squeue`, `sbatch`, `skill` 명령어들에 대해 알아보고 사용해 보자. (보고서에는 포함하지 말 것)
- 과도하게 많은 작업을 스케줄러에 제출해서 다른 학생들의 실험을 방해하지 않도록 하자.

보고서에는 다음 내용이 들어가면 좋다.

- 병렬화 방법에 대한 설명
- Single precision 기준 NVIDIA GeForce GTX 2080 GPU의 Theoretical peak FLOPS는 약 10 TFLOPS이다. 어떻게 계산하는가? 작성한 프로그램은 theoretical peak FLOPS 에 비해 어느 정도의 성능이 나오는가? 느리다면 그 이유에 대한 분석.

채점 기준은 다음과 같다. 이전 과제에 비해 보고서의 비중이 줄고 성능의 비중이 늘어났음에 유의하라.

보고서 (20%)

정확성 (40%) 4096 이하의 임의의 M, N, K에 대해서 `-v` 옵션을 통한 validation을 통과해야 한다.

성능 (40%) $M = N = K = 8192$ 옵션을 주고 실행했을 때, 600 GFLOPS를 넘으면 만점. (peak 성능의 6%) 그 이하는 비율에 따라 점수를 부여한다. (e.g., 300 GFLOPS인 경우 성능 점수의 50%를 부여) 답이 틀린 경우 0점.

2 문제 2 (10%)

문제 1에서 측정하지 않았던 부분(platform, device, context, queue, kernel 생성, 프로그램 빌드, 메모리 전송 등)의 실행 시간을 측정해보고, 유의미하게 오래 걸리는 것이 있는지 확인하여 보고서에 결과를 논의하라. 문제 2에 대한 소스 코드는 제출하지 않아도 된다.

3 제출 방법

- 조교 메일(`jinpyo@aces.snu.ac.kr`)로 보고서를 포함한 모든 파일을 하나의 파일(e.g., .zip, .tar.gz)로 압축 후 첨부하여 제출한다.
- 메일 제목은 [mc2021] 계정이름_HW3으로 한다. (e.g., [mc2021] mc99_HW3)
- 첨부파일명은 계정이름_HW3.확장자으로 한다. (e.g., mc99_HW3.zip, mc99_HW3.tar.gz)

- 제출한 메일은 기계적으로 처리되므로 위의 내용을 지키지 않을 시 누락될 수 있으니 잘 지켜주시기 바랍니다.
- Grace day를 사용하고자 하는 경우에는 메일 내용에 이를 반드시 포함한다.