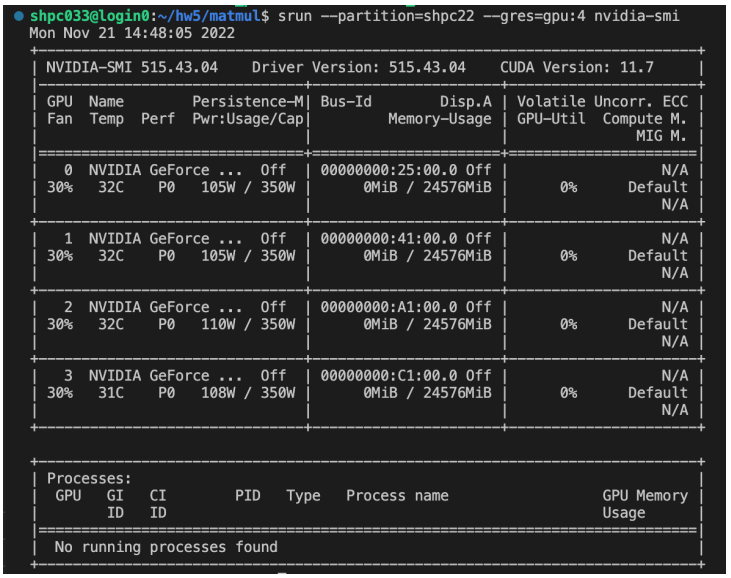
SHPC22 HW5
2022-28981 Jungsoo Kim

1.
(a)-1.
shpc22 partition을 실행함에 노드 1개당 gpu4개를 할당하여 gpu의 상태 및 기본정보(드라이버 버전, GPU의 팬의 성능, 현재 돌고있는 프로세스 등)를 보여준다.



(a)-2.
shpc22 partition을 실행함에, 노드에 연결된 각GPU들에 대한 보다 상세한 정보(각 GPU의 스펙)를 보여준다. 노드 당 4개이므로 4개의 gpu에 대한 정보가 나온다. (ex. 000000:25:00.0slot에 연결된 gpu의 이름, 브랜드, 아키텍쳐, 기술스펙 등)

```
shpc033@login0:~/hw5/matmul$ srun --partition=shpc22 --gres=gpu:4 nvidia-smi -q

==============NVSMI LOG==============

Timestamp                                 : Mon Nov 21 14:50:48 2022
Driver Version                            : 515.43.04
CUDA Version                              : 11.7

Attached GPUs                             : 4
GPU 00000000:25:00.0
    Product Name                          : NVIDIA GeForce RTX 3090
    Product Brand                         : GeForce
    Product Architecture                  : Ampere
    Display Mode                          : Disabled
    Display Active                        : Disabled
    Persistence Mode                      : Disabled
    MIG Mode
        Current                           : N/A
        Pending                           : N/A
    Accounting Mode                       : Disabled
    Accounting Mode Buffer Size           : 4000
    Driver Model
        Current                           : N/A
        Pending                           : N/A
    Serial Number                         : N/A
    GPU UUID                              : GPU-31c5c389-72cd-e985-4951-f64739522442
    Minor Number                          : 1
    VBIOS Version                         : 94.02.42.40.34
    MultiGPU Board                        : No
    Board ID                              : 0x2500
    GPU Part Number                       : N/A
    Module ID                             : 0
    Inforom Version
        Image Version                     : G001.0000.03.03
        OEM Object                        : 2.0
        ECC Object                        : N/A
        Power Management Object           : N/A
    GPU Operation Mode
        Current                           : N/A
        Pending                           : N/A
    GSP Firmware Version                  : N/A
    GPU Virtualization Mode
        Virtualization Mode               : None
```

(a)-3.

shpc22 partition에 대해, srun으로 할당된 gpu노드에 대해 opencl의 플랫폼 정보(플랫폼이름, vendor, 버전 등)과 디바이스 parameters를 보여준다.

https://github.com/simleb/clinfo

```
shpc033@login0:~/hw5/matmul$ srun --partition=shpc22 --gres=gpu:4 clinfo
Number of platforms                               1
  Platform Name                                   NVIDIA CUDA
  Platform Vendor                                 NVIDIA Corporation
  Platform Version                                OpenCL 3.0 CUDA 11.7.57
  Platform Profile                                FULL_PROFILE
  Platform Extensions                             cl_khr_global_int32_base_atomics cl_khr_global_int32_extended_atomics cl_khr_local_int32_base_atomics cl_khr_local_int32_extended_atomics cl_khr_fp64 cl_khr_3d_image_writes cl_khr_byte_addressable_store cl_khr_icd cl_khr_gl_sharing cl_nv_compiler_options cl_nv_device_attribute_query cl_nv_pragma_unroll cl_nv_copy_opts cl_nv_create_buffer cl_khr_int64_base_atomics cl_khr_int64_extended_atomics cl_khr_device_uuid cl_khr_pci_bus_info cl_khr_external_semaphore cl_khr_external_memory cl_khr_external_semaphore_opaque_fd cl_khr_external_memory_opaque_fd
  Platform Host timer resolution                  0ns
  Platform Extensions function suffix             NV

  Platform Name                                   NVIDIA CUDA
Number of devices                                 4
  Device Name                                     NVIDIA GeForce RTX 3090
  Device Vendor                                   NVIDIA Corporation
  Device Vendor ID                                0x10de
  Device Version                                  OpenCL 3.0 CUDA
  Driver Version                                  515.43.04
  Device OpenCL C Version                         OpenCL C 1.2
  Device Type                                     GPU
  Device Topology (NV)                            PCI-E, 25:00.0
  Device Profile                                  FULL_PROFILE
  Device Available                                Yes
  Compiler Available                              Yes
  Linker Available                                Yes
  Max compute units                               82
  Max clock frequency                             1695MHz
  Compute Capability (NV)                         8.6
  Device Partition                                (core)
    Max number of sub-devices                     1
    Supported partition types                     None
    Supported affinity domains                    (n/a)
  Max work item dimensions                        3
  Max work item sizes                             1024x1024x64
  Max work group size                             1024
  Preferred work group size multiple              32
  Warp size (NV)                                  32
  Max sub-groups per work group                   0
  Preferred / native vector sizes
    char                                                 1 / 1
    short                                                1 / 1
    int                                                  1 / 1
    long                                                 1 / 1
    half                                                 0 / 0        (n/a)
    float                                                1 / 1
    double                                               1 / 1        (cl_khr_fp64)
  Half-precision Floating-point support           (n/a)
  Single-precision Floating-point support         (core)
    Denormals                                     Yes
    Infinity and NANs                             Yes
    Round to nearest                              Yes
    Round to zero                                 Yes
    Round to infinity                             Yes
    IEEE754-2008 fused multiply-add               Yes
    Support is emulated in software               No
    Correctly-rounded divide and sqrt operations  Yes
  Double-precision Floating-point support         (cl_khr_fp64)
```

(b)
NVIDIA GeForce RTX 3090, 4개

(c)
24576 MiB

(d)
350W, 2100MHz

(e) max workitem dimension: 3, max work item size : 1024x1024x64, max work group size 1024

2.
Reference: https://yunmorning.tistory.com/37

(a) 애초에 off-chip memory에 접근하는 횟수를 num_tiles만큼 줄인다.
그리고 vector type(float8)을 이용해 병렬화되는 work per thread수를 늘린다.
(integrity handling:

(b)
matmul_initialize: opencl 기본 세팅을 해주고, command queue생성, kernel program 컴파일, 원 데이터를 받아갈 buffer 등을 생성해준다.

(i) clGetPlatformIDs: open cl 플랫폼 id 정보 얻어오기
(ii) clGetDeviceIDs: openCL device얻어오기
(iii) clCreateContext: opencl context만들기
(iv) clCreateCommandQueue: opencl command queue생성
(v) create_and_build_program_with_source: kernel.cl 컴파일시키기
(vi) clCreateKernel: 컴파일된 kernel.cl에서 함수를 뽑아내기 (sgemm, padding, padding제거 함수를 뽑아냈음)
(vii) clCreateBuffer: 데이터를 받아갈 버퍼 생성 (read_only or read_write정하기)

matmul: kernel.cl의 kernel함수들을 실질적으로 실행하는 함수. 행렬곱의 실질적인 연산을 수행하게 해준다.
ex. A,B가 들어오면, A,B를 워크그룹사이즈에 맞추기 위해 크기가 안맞으면 패딩을 시키고, 패딩을 시킨 후 kernel.cl의 sgemm을 실행해 행렬곱을 수행하고 그렇게 결과로 나온 행렬을 패딩된 부분을 지워서 내보내야 한다

(i) clEnQueueWriteBuffer: GPU버퍼에 원 데이터(행렬A,B)를 받아쓴다.
(ii) clSetKernelArg: 각 kernel의 함수에 들어갈 argument들을 해당 argument가 들어갈 버퍼이름과 함께 적어 argument setting을 한다. (argument순서대로 하나씩 쓴다)
(iii) clEnqueueNDRangeKernel: kernel함수들을 실행 (즉, 실행될커널을 command queue에 추가)
(iv) clEnqueueReadBuffer: GPU에서 만든 특정 kernel함수 실행 결과를 CPU로 읽어온다.
(v) clFinish: 그전에 실행한 writebuffer가 끝날 때까지 기다린다.

matmul_finalize: 연산 완료 후, gpu에서 생성된 객체와 opencl component에 대한 memory를 release한다.

(i) clReleaseMemObject: 메모리 오브젝트 release
(ii) clReleaseCommandQueue: CommandQueue release
(iii) clReleaseProgram: opencl 프로그램 release
(iv) clReleaseKernel: 커널 release

(c)

- Buffer Create하는 방식
  raw 행렬을 담을 buffer를 만들 때 read만 하게 되는 A,B의 buffer들의 경우엔

```
a_d = clCreateBuffer(context, CL_MEM_READ_ONLY, M * K * sizeof(float),
NULL,&err);
```

  로 바꿔서 실행하면 조금 더 성능이 좋아진다.
  (생성한 buffer가 할당되는 메모리의 역할 한정)
  (대략 ./run_performance.sh를 돌릴 때 10-20GFLOP 좋아짐)
- workgroup 사이즈
  workgroup사이즈를 32->64로 바꿨을 때 즉, workgroup내 workitem수를 늘렸더니 성능이 거의 600GFLOPS향상됐다. 적당한 범위를 찾는 게 중요할 것 같다.

```
shpc033@login0:~/hw5/matmul$ make clean && make && ./run_performance.sh
rm -rf main util.o matmul.o
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/loc
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/loc
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/loc
n.c util.o matmul.o  -lm -pthread -lOpenCL -o main
Options:
  Problem size: M = 8192, N = 8192, K = 8192
  Number of iterations: 10
  Print matrix: off
  Validation: on

Initializing... done!
Initializing OpenCL...
Detected OpenCL platform: NVIDIA CUDA
Detected OpenCL device: NVIDIA GeForce RTX 3090
Calculating...(iter=0) 0.359479 sec
Calculating...(iter=1) 0.352542 sec
Calculating...(iter=2) 0.367944 sec
Calculating...(iter=3) 0.372341 sec
Calculating...(iter=4) 0.357833 sec
Calculating...(iter=5) 0.358509 sec
Calculating...(iter=6) 0.370584 sec
Calculating...(iter=7) 0.345169 sec
Calculating...(iter=8) 0.377312 sec
Calculating...(iter=9) 0.342105 sec
Validating...
Result: VALID
Avg. time: 0.360382 sec
Avg. throughput: 3050.963936 GFLOPS
```

```
shpc033@login0:~/hw5/matmul$ make clean && make && ./run_performance.sh
rm -rf main util.o matmul.o
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/lo
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/lo
gcc  -O3 -Wall -march=native -mavx2 -mfma -fopenmp -mno-avx512f -I/usr/lo
n.c util.o matmul.o  -lm -pthread -lOpenCL -o main
Options:
  Problem size: M = 8192, N = 8192, K = 8192
  Number of iterations: 10
  Print matrix: off
  Validation: on

Initializing... done!
Initializing OpenCL...
Detected OpenCL platform: NVIDIA CUDA
Detected OpenCL device: NVIDIA GeForce RTX 3090
Calculating...(iter=0) 0.304553 sec
Calculating...(iter=1) 0.289551 sec
Calculating...(iter=2) 0.302078 sec
Calculating...(iter=3) 0.312521 sec
Calculating...(iter=4) 0.335040 sec
Calculating...(iter=5) 0.289113 sec
Calculating...(iter=6) 0.310338 sec
Calculating...(iter=7) 0.313063 sec
Calculating...(iter=8) 0.290769 sec
Calculating...(iter=9) 0.295664 sec
Validating...
Result: VALID
Avg. time: 0.304269 sec
Avg. throughput: 3613.616545 GFLOPS
```