

Homework 2

CSCI 680 GPU Architectures

Yu Chen

1. **Is your program working correctly and TEST passed?**

Yes (I have commented the verifying msgs):

```
ychen39@bg4:~/code/gpu/hw2-files> ./vector_add

Setting up the problem...0.000033 s
  size 0f vector: 1000 x 0
  Allocating device variables...0.317253 s
Copying data from host to device...0.003782 s
Launching kernel...0.002178 s
Copying data from device to host...0.000027 s
Verifying results...
TEST PASSED

ychen39@bg4:~/code/gpu/hw2-files> █
```

Figure 1: TEST PASSED

2. **How many total thread blocks do we use?**

Block size is 256 and total threads' number we need is 1000 so that block number is $\lceil \frac{1000}{256} \rceil = 4$.

3. **Are all thread blocks full? That is, do all threads in thread blocks have data to operate on?**

No. The first three blocks are full which run $256 * 3 = 768$ threads totally. The last block run $1000 - 768 = 232$ threads and its size is 256 thus $256 - 232 = 24$ threads have no data

4. **Use nvprof to analyze the results. Paste your output in the report and discuss at least three observations.**

Observations:

- There are two categories of record, one is "GPU activities" that profiles GPU side's activities. Another one is "API calls" that profiles driver side's activities.
- "cudaMalloc" is the most time consuming call (161.71ms). "cuDeviceGetAttribute" is called most times (384 calls).
- With the command "nvprof -print-api-trace ./vector_add", we can get API calls' orders and each call's time consuming.

```

TEST PASSED

==7041== Profiling application: ./vector_add
==7041== Profiling result:
   Type  Time(%)   Time     Calls   Avg       Min       Max  Name
GPU activities:  51.60%  4.6400us      2  2.3200us  1.5680us  3.0720us  [CUDA memcpy HtoD]
                31.32%  2.8160us      1  2.8160us  2.8160us  2.8160us  VecAdd(int, float const
*, float const *, float*)
                17.08%  1.5360us      1  1.5360us  1.5360us  1.5360us  [CUDA memcpy DtoH]
API calls:      95.55%  503.14ms      3  167.71ms  8.0350us  503.11ms  cudaMalloc
                1.84%  9.6654ms      3  3.2218ms  7.4900us  9.6399ms  cudaFree
                1.11%  5.8619ms      4  1.4655ms  3.8920us  2.1912ms  cudaDeviceSynchronize
                0.86%  4.5468ms     384  11.840us    173ns  1.5507ms  cuDeviceGetAttribute
                0.42%  2.2002ms      3  733.41us  26.970us  2.1274ms  cudaMemcpy
                0.14%  745.95us      4  186.49us  149.04us  225.04us  cuDeviceTotalMem
                0.07%  351.03us      4  87.756us  77.930us  112.94us  cuDeviceGetName
                0.01%  45.310us      1  45.310us  45.310us  45.310us  cudaLaunchKernel
                0.00%  13.020us      4  3.2550us  2.3810us  5.1920us  cuDeviceGetPCIBusId
                0.00%  4.5520us      8    569ns    192ns  1.4100us  cuDeviceGet
                0.00%  4.2700us      3  1.4230us  438ns  3.3170us  cuDeviceGetCount
                0.00%  1.4630us      4    365ns    270ns   429ns  cuDeviceGetUuid

ychen39@bg4:~/code/gpu/hw2-files>

```

Figure 2: nvprof ./vector_add

```

ychen39@bg4:~/code/gpu/hw2-files> nvprof --print-api-trace ./vector_add

Setting up the problem...0.000131 s
    size Of vector: 1000 x 0
    Allocating device variables...==7294== NVPROF is profiling process 7294, command: ./ve
0.720298 s
Copying data from host to device...0.004074 s
Launching kernel...0.002158 s
Copying data from device to host...0.000054 s
Verifying results...
TEST PASSED

==7294== Profiling application: ./vector_add
==7294== Profiling result:
   Start  Duration  Name
147.79ms  4.0110us  cuDeviceGetPCIBusId
181.13ms  3.8760us  cuDeviceGetPCIBusId
187.21ms  1.8220us  cuDeviceGetPCIBusId
193.25ms  1.7640us  cuDeviceGetPCIBusId
199.41ms  1.1580us  cuDeviceGetCount
199.41ms    276ns  cuDeviceGetCount
199.75ms  12.619us  cuDeviceGet
199.76ms    762ns  cuDeviceGetAttribute
199.77ms    344ns  cuDeviceGetAttribute
199.77ms    376ns  cuDeviceGetAttribute
199.78ms    242ns  cuDeviceGet
199.78ms    252ns  cuDeviceGetAttribute
199.78ms    185ns  cuDeviceGetAttribute
199.78ms    251ns  cuDeviceGetAttribute
199.79ms    195ns  cuDeviceGet
199.79ms    253ns  cuDeviceGetAttribute
199.79ms    217ns  cuDeviceGetAttribute
199.79ms    253ns  cuDeviceGetAttribute
199.99ms  1.4390us  cuDeviceGet
200.00ms    454ns  cuDeviceGetAttribute
200.00ms    203ns  cuDeviceGetAttribute
200.00ms    291ns  cuDeviceGetAttribute
200.04ms    390ns  cuDeviceGetCount
200.04ms    204ns  cuDeviceGet
200.05ms  111.55us  cuDeviceGetName
200.16ms  226.17us  cuDeviceTotalMem
200.38ms    447ns  cuDeviceGetAttribute
200.39ms    188ns  cuDeviceGetAttribute

```

Figure 3: nvprof --print-gpu-trace ./vector_add