Homework 2
CSCI 680 GPU Architectures

Yu Chen

1. **Is your program working correctly and TEST passed?**
   Yes (I have commented the verifying msgs):

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.

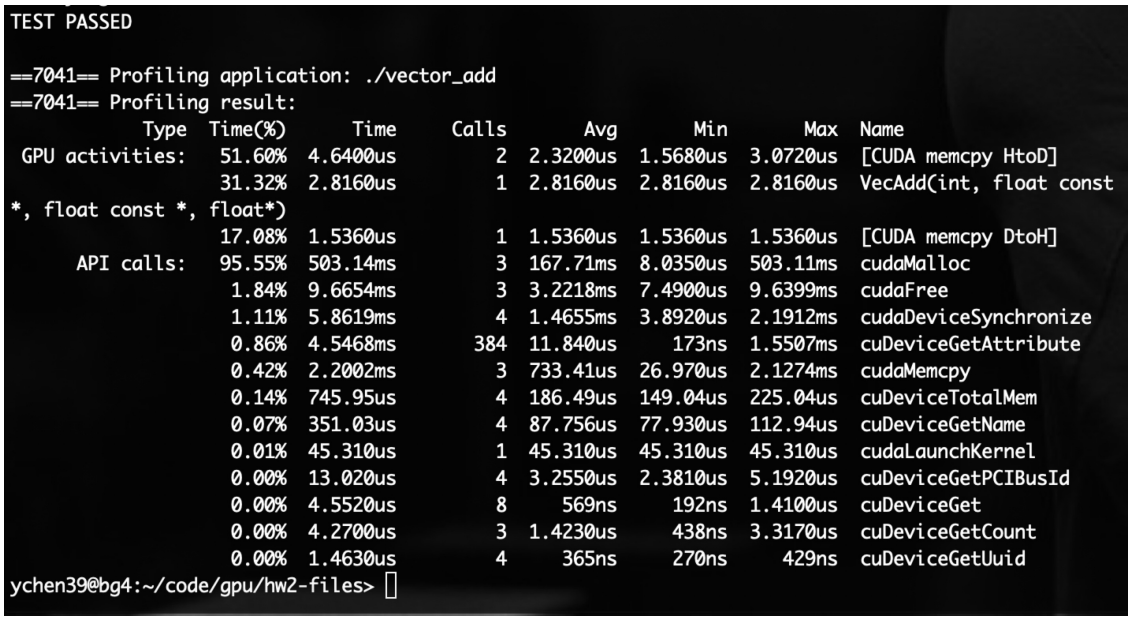

Figure 2: nvprof ./vector_add

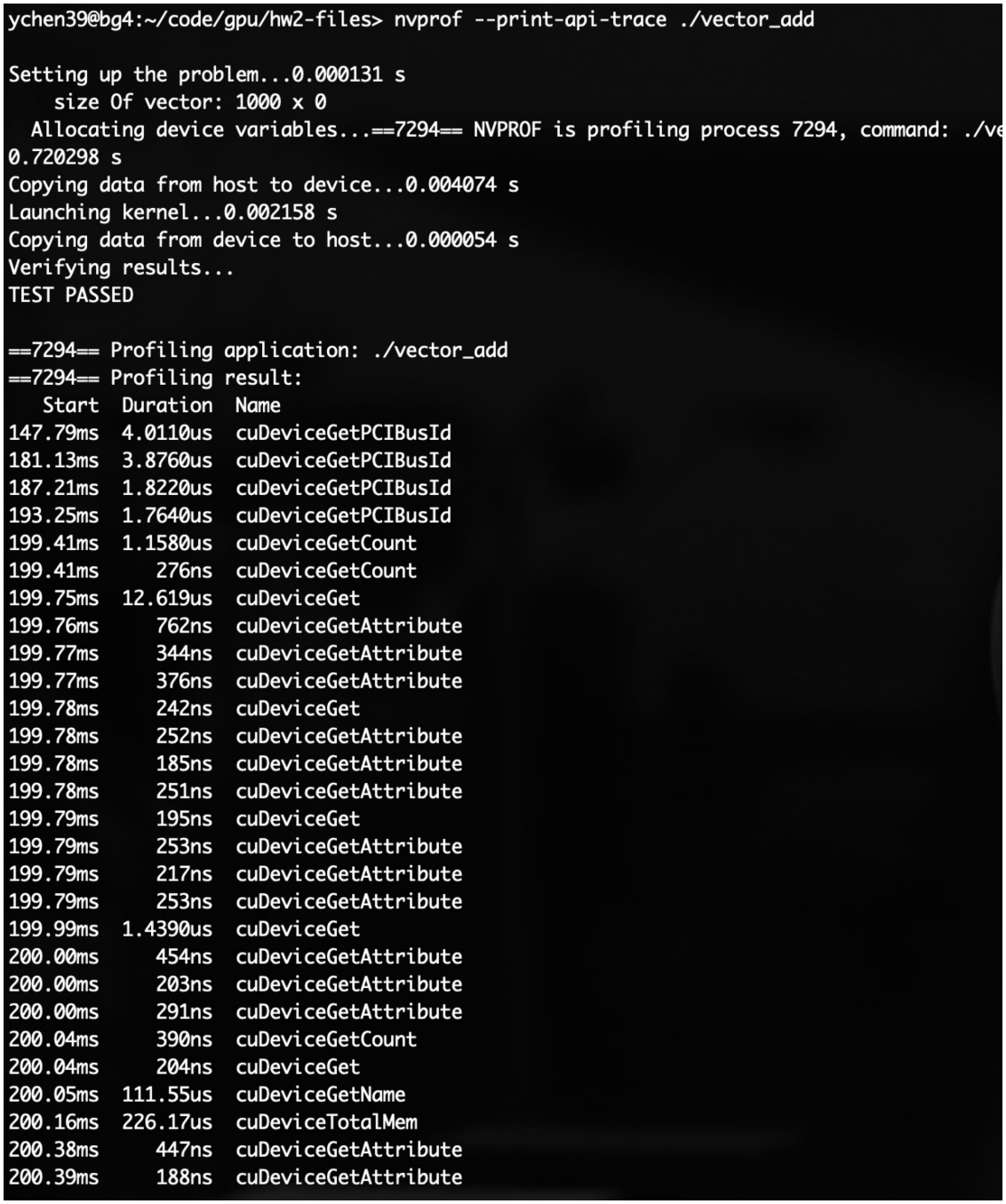

Figure 3: nvprof –print-gpu-trace ./vector_add